

Jalostustarkastus – lokaali sovellus tallentajille CASE SML

Iris Erkkilä

Tekijä tai tekijät Iris Erkkilä	Ryhmätunnus tai aloitusvuosi 2009
Raportin nimi Jalostustarkastus – lokaali sovellus tallentajille CASE SML	Sivu- ja liitesivumäärä 20 + 49
Opettajat tai ohjaajat Jyri Partanen	
<p>Projekti käynnistettiin Suomen Mäyräkoiraliiton tarpeesta uusia vanha tietokanta, sekä kirjaus- ja tallennustyökalu mäyräkoirien jalostustarkastusta varten. Vanha järjestelmä on tullut elinkaarensa päähän ja sen tilalle päätettiin kehittää kokonaan uusi selainpohjainen järjestelmä. Toteutettiin jalostustarkastajien koneille lokaali sovellus omalla tietokannalla. Tähän toteutustapaan päädyttiin siksi, että tarkastuksien suorituspaikoilla ei välttämättä ole Internet verkkoa saatavilla.</p> <p>Tavoitteina oli toteuttaa selkeä ja toimiva järjestelmä, joka tulisi helpottamaan jalostustarkastajien työtä. Järjestelmä toimii lokaalisti tallentajan kannettavalla tietokoneella käyttäen lokaalia palvelinta ja tietokantaa. Työ käsittelee vain jalostustarkastajien uutta kirjaus- ja tallennustyökalua, sekä lokaalia tietokantaa. Työstä on rajattu pois vanhan tietokannan migratointi uuteen.</p> <p>Työ oli jaettu kahteen osaan, suunnitteluvaiheeseen ja toteutukseen. Työn suunnittelun tuloksena syntyi suunnittelukuvasto. Työn toteutuksessa suosittiin avoimen lähdekoodin ohjelmia ja muita mahdollisia ilmaisohjelmia, sillä projektille ei ollut varattuna budjettia. Työ toteutettiin vuoden 2012 aikana. Jalostustarkastus-projekti käynnistettiin kesäkuussa 2012 ja se saatettiin päätökseen marraskuussa 2012.</p> <p>Projektissa käytettiin ketterän kehittämisen työmallia Scrum, joka soveltuu pienten projektien kehitykseen. Työn laatu ja vaatimusten toteutuminen varmistettiin järjestelmätestauksilla, jotka suoritti toimeksiantaja.</p> <p>Työn tuloksena syntyi, asetetut vaatimukset täyttävä, lokaalisovellus. Ketterän kehittämisen työmalli soveltui projektin kehittämiseen hyvin sen keveyden ja nopean toteutustahdin ansiosta. Erillistä määrittelyvaihetta ei koettu tarpeelliseksi, vaan dokumentaatioista toteutettiin kevyemmät versiot.</p>	
Asiasanat Ohjelmistokehitys, verkko-ohjelmointi, PHP	

Authors Iris Erkkilä	Group or year of entry 2009
The title of thesis Dogs' breeding examination – a local application for breeding auditors CASE: Suomen Mäyräkoiraliitto	Number of pages and appendices 20 + 49
Supervisor(s) Jyri Partanen	
<p>This thesis project was started due to the need of the Suomen Mäyräkoiraliitto (SML, the Finnish Dachshund Association) to renew the old database and the recording tool for dogs' breeding examination. The old system had come to its end and it was decided to develop a new web based system to replace it.</p> <p>The aim of this thesis project was to create a clear and workable system that would facilitate the work of the auditors of the dogs' breeding examinations. The project included a new recording tool and a locale database. The data migration from the old database to the new one was left out from this project.</p> <p>The application was installed in the auditors' laptops. This was the most appropriate way to implement the application because there might not always be an internet connection available where the examinations are arranged.</p> <p>The project was derived into two phases: design and implementation. A design catalog was made in the design phase. In the implementation of the work, open source software and other free software were favored because the project did not have a budget. The work was carried out during 2012. The project was launched in June 2012 and it was completed in November 2012.</p> <p>An agile development working model Scrum was used because it is suitable for small sized projects like this one. The quality of the work and the implementation of the requirements were confirmed with system testing. Those tests were conducted by the client.</p> <p>The result met the requirements set for of the local application. Agile software development suited well to this project because of its rapid implementation pace. A separate definition phase was considered unnecessary. Instead of that, the documentation was carried out with lighter versions.</p>	
Key words Software development, network programming, PHP	

Sisällys

1	Johdanto	1
2	Kohteen esittely.....	3
2.1	Jalostustarkastus	3
2.2	Jalostustarkastuksen järjestäminen	3
3	Ohjelmistokehittämisen menetelmät	4
3.1	Keskeisimmät mallit.....	4
3.1.1	Vesiputousmalli.....	4
3.1.2	RUP	6
3.1.3	Scrum	7
3.2	Kuvauskäytännöt.....	9
4	Projektin tavoitteet.....	9
4.1	Valitut työtavat ja -välineet.....	9
4.2	Ketterän kehityksen menetelmät.....	11
4.3	Projektin tehtävä.....	12
4.4	Tavoitteet ja lopputulokset	12
4.5	Valmis ohjelma – kenelle tulee	13
5	Jalostustarkastus–projekti.....	14
5.1	Projektin aloitus.....	14
5.2	Määrittäminen.....	14
5.3	Suunnittelu	14
5.4	Toteutus.....	16
5.5	Testaus ja tulosten seuranta	17
5.6	Projektin päättäminen.....	17
6	Tulosten esittely.....	18
7	Pohdinta	18
7.1	Tulokset	18
7.2	Kustannusarviot ja työajanseuranta	19
7.3	Oman oppimisen arviointi	19
7.4	Jatkokehitysehdotukset.....	20
	Lähteet	21

Liitteet.....	1
Liite 1. Product Backlog.....	1
Liite 2. Suunnittelukuvasto.....	2
Liite 3. Aikataulu toteutma.....	1
Liite 4. Testauslomakkeet.....	1

1 Johdanto

Projekti käynnistettiin Suomen Mäyräkoiraliiton tarpeesta uusia vanha tietokanta, sekä tallennustyökalu mäyräkoirien jalostustarkastusta varten. Vanha järjestelmä on tullut elinkaarensa päähän ja se täytyi päivittää tähän päivään. Suomen Mäyräkoiraliitto on käynnistänyt oman maykkynet -projektinsa, jossa uusitaan maykkynet-tietokanta ja siihen julkinen selailunäkymä. Tähän liittyen käynnistettiin myös jalostustarkastus - projekti. Tämän projektin tehtäväksi on rajattu lokaalin sovelluksen suunnittelu ja toteutus jalostustarkastajille, jotka uuden sovelluksen avulla voivat suorittaa SML järjestämiä mäyräkoirien jalostustarkastuksia.

Projektin päätavoitteena on tuottaa Suomen Mäyräkoiraliiton jalostustoimikunnalle lokaalisovellus jalostustarkastusten suorittamiseen. Tavoitteena on selkeyttää jalostustarkastusraportin ulkoasua ja luettavuutta, sekä tarkastuspaikalla suoritettavaa kirjausta.

Ketterä kehitys ja Scrum ovat suuren suosion saaneet ohjelmistokehittämisen menetelmät, joita erityisesti suosivat pienet projektit ja projektiryhmät. Sovelluskehityksessä tärkeintä kuitenkin on soveltaa olemassa olevia malleja omaan projektiin sopivaksi, jotta saadaan aikaiseksi paras mahdollinen tulos. Scrumin katsottiin soveltuvan tämän projektin kehitysmenetelmäksi sen nopeiden työvaiheiden ja kevytrakenteisuuden vuoksi. Scrumissa sovellusta testataan useammin ja mahdolliset ongelmat saadaan korjattua ajoissa.

Työn tuloksena muodostui selainpohjainen käyttöliittymä, joka käyttää omaa lokaalia tietokantaa tietojen väliaikaista tallennusta varten. Selainpohjaiset käyttöliittymät ovat nousseet hyväksi vaihtoehdoksi työpöytäsovelluksen rinnalle. Selainpohjaisten sovellusten hyvänä puolena on se, että ne eivät ole riippuvaisia käyttöjärjestelmästä.

Työ alkaa kohteen esittelyllä. Suomen Mäyräkoiraliitto esitellään hyvin karkealla tasolla ja jalostustarkastus ja sen järjestäminen esitellään hieman yksityiskohtaisemmin, sillä siihen tämän työn sisältö on kohdistettu. Luvussa 3 esitellään erilaisia ohjelmistokehittämisen malleja, mitä muita mahdollisia tapoja on tarjolla kuin Scrum. Luvussa 4 kuvataan projektin tavoitteet, käytetyt menetelmät ja työvälineet ja tehtävät. Tämän jälkeen

kerrotaan projektin kulusta ja työvaiheet esitellään. Luvussa 6 esitetään projektin tulokset ja lopuksi on vertailtu työn toteutumista tehtyjen valintojen osalta, tarkastellaan oppimisprosessia ja esitellään jatkokehitysehdotukset.

2 Kohteen esittely

Suomen Mäyräkoiraliitto on Suomen Kennelliiton hyväksymä rotujärjestö, joka suorittaa mm. jalostustarkastuksia mäyräkoirille. Projekti käsittelee SML:n jalostustarkastusprosessia ja sen kehittämistä, eikä sen vuoksi muita osa-alueita esitellä tässä työssä. (Suomen Mäyräkoiraliitto SML ry a.)

2.1 Jalostustarkastus

Jalostustarkastuksessa kerätään mäyräkoirista tietoja, joista selviää tarkempaa tietoa koirasta ja ominaisuuksista, joita kyseinen koira mahdollisesti periyttää. Sen toimittaa Suomen Mäyräkoiraliitto. Jalostustarkastuksessa koira punnitaan ja siitä mitataan seuraavat kohteet: pään pituus, rinnan ympäryys, säkäkorkeus, maavara, rungon syvyys, rungon pituus. Koirasta otettuja mittoja verrataan rotumääritelmän esittemiin arvoihin ja näiden vertailujen avulla tarkistetaan, missä määrin koira täyttää rotumääritelmän. Tarkastuksessa täytetään myös arviointilomake, joka pitää sisällään arvioitavat kohteet. Tarkastajan on arvioitava koirasta lomakkeella ilmoitetut kohteet. Nämä kohteet ja niiden arviointi on kuvattu rotumääritelmässä.

Mittausten ja kohteiden arvioinnin jälkeen annetaan koiralle loppullinen lausunto tai ns. lopputulos. Lopputulos valitaan ennalta määrättyjen hyväksymis- ja hylkäämisperiaatteiden mukaan. Pääsääntöisesti samalle koiralle tehdään tarkastus vain kerran, mutta tietyissä tapauksissa koira voidaan tarkastaa uudelleen. Esimerkiksi, jos koira on alle 15 kk vanhana jalostustarkastettu, on tulos silloin epävirallinen ja tarkastus voidaan suorittaa myöhemmin uudelleen. (Suomen Mäyräkoiraliitto SML Ry b.)

2.2 Jalostustarkastuksen järjestäminen

Koiralleen jalostustarkastuksen haluava pyytää tarkastuksen jalostustoimikunnalta, joka päättää suoritetaanko tarkastus ja päättää ajankohdan jolloin tarkastus suoritetaan. Menettelytapa edesauttaa henkilötyöväimän ja ajanhallinnan jakamista parhaimmalla mahdollisella tavalla ja tarkastusten pysymistä aikataulussa. ”Jalostustoimikunta tuo paikalle edustajansa, tietokoneen, printterin, tarkastuslomakkeet, vaa'an ja mitat kuten aieminkin.” (Suomen Mäyräkoiraliitto SML Ry b.)

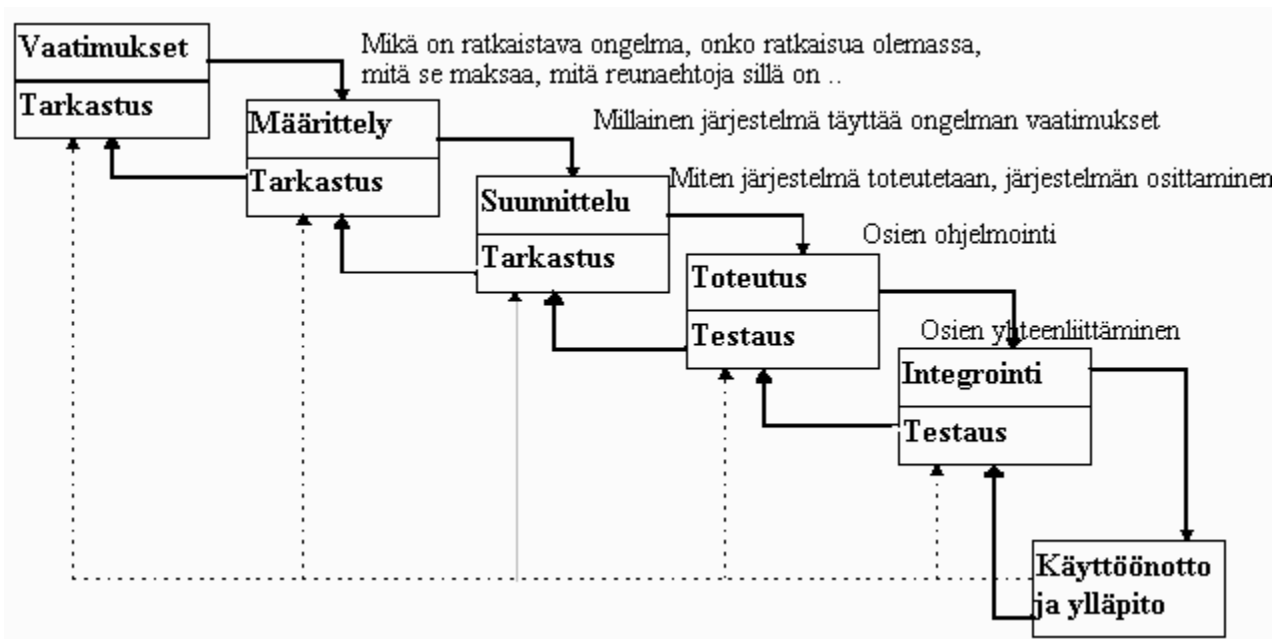
3 Ohjelmistokehittämisen menetelmät

3.1 Keskeisimmät mallit

Ohjelmiston kehitystyö jaetaan eri vaiheisiin ja näitä jakoja tehdään eri mallien avulla. Ohjelmistokehityksen keskeisimmät vaiheet kaikissa malleissa on yleensä samat: määrittely, suunnittelu, toteutus ja testaus. Tässä luvussa esitellään muutama eri vaihejakomalli. (I. Haikala, J. Märijärvi, 2004, s35.)

3.1.1 Vesiputousmalli

Perinteisin malleista on vesiputousmalli, jonka rakenne on esiteltynä kuviossa 1. Vesiputousmallissa vaatimukset kerätään asiakkaan esittämistä vaatimuksista järjestelmän suhteen. Tämän vaiheen tulee kertoa miksi järjestelmä tehdään ja mitä sen tulee täyttää. Jos asiakkaan tarpeita ei ole oikein ymmärretty tai niitä ei ole kunnolla saatu selvitettyä, voi tuloksena olla huono järjestelmä. Määrittelyvaiheessa kirjataan vaatimusten pohjalta tarkat toimintaperiaatteet ja käyttötarkoitus sekä sovelluksen ympäristö, toimijat ja luokkamallit. Järjestelmän suunnittelu ja toteutus vaiheita ei usein toteuteta kaikkea kerralla vaan ne jaetaan usein muutaman käyttötapauksen suuruisiin toteutusvaiheisiin, iteraatioihin. Suunnitteluvaihe on enemmän teknistä suunnittelua siitä kuinka varsinainen järjestelmä toteutetaan. Suunnitteluvaiheen tuloksena syntyy usein suunnittelukuvasto. Toteutusvaiheessa toteutetaan suunniteltu järjestelmä. Iteraatioissa toteutetut osat yhdistetään integtoinrivaiheessa. Elinkaari päättyy järjestelmän käyttöönoton ja ylläpidon jälkeen, kun ohjelman ylläpito tietyn ajan kuluessa lopetetaan. (Haikala & Märijärvi. 2004.)



Kuvio 1. Vesiputousmallin elinkaari (Haikala & Märijärvi. 2004: 36)

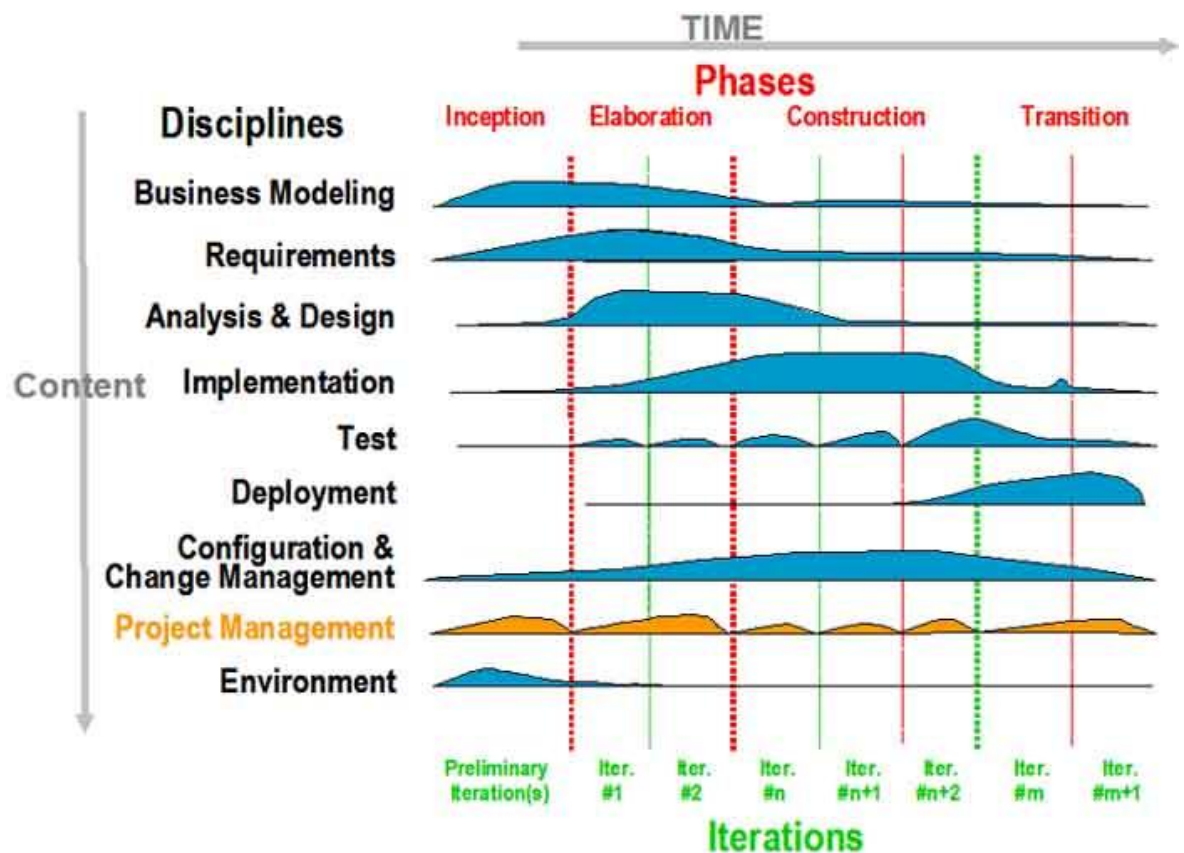
Jokaisesta vaiheesta kirjoitetaan kattava dokumentaatio. Dokumentaation suuren määrän vuoksi se vaatii paljon aikaa, resursseja ja varoja, mutta kattava dokumentaatio auttaa projektin ulkopuolisia tai uusia tahoja ymmärtämään järjestelmän sisällön. Tarkka ja hyvin kuvaava dokumentaatio on tärkeä, silloin kun on kyseessä suuri kehityshanke. (Sommerville, I, 2007, s66-67.)

Usein vesiputousmallissa valmiiksi tehty vaihe jäädytetään, eikä sitä enään muuteta, sillä muutokset maksavat. Tästä johtuen myöhemmässä vaiheessa ilmaantuvat ongelmat kierretään, jätetään myöhempään vaiheeseen tai niiden vain annetaan olla. Suurimpana ongelman onkin mallin joustamattomuus

Vesiputousmalli voi olla liian työläs ja raskas pienelle projektille, mutta soveltuu hyvin suuriin tuotekehityshankkeisiin, joissa projektiryhmä ei välttämättä työskentele tiiviisti yhdessä tai projektin eri vaiheilla ja osa-alueilla on mahdollisesti eri kehittäjät. Tällöin tarkka dokumentaatio antaa selvän kuvan, että kaikki osaavat tehdä yhtenäistä työtä. (Sommerville, I, 2007, s66-67.)

3.1.2 RUP

RUP (IBM Rational Unified Process) on moderni ohjelmiston prosessimalli, joka on kehitetty käyttämään hyväkseen UML notaatioita. Kuviossa 2 on yksi esimerkki RUP:n työvaiheista sen elinkaaren aikana. RUP on iteratiivista kehitystä ja siinä on ideana suuret toimitus määrät. Sitä käytetään yleensä isoissa ohjelmisto järjestelmissä, eikä se prosessina välttämättä sovellus kaikkiin kehitystapoihin. RUP kuvataan usein kolmella tavalla: dynaaminen näkökulma, staattinen näkökulma ja harjoitus näkökulma. (Sommerville, I, 2007, s82.)



Kuvio 2. RUP ohjelmistokehittämisen elinkaari (Cottrell, W. 2004)

Tässä esitellyssä RUP mallissa on neljä työvaihetta: aloitus, suunnittelu, rakentaminen/toteutus ja käyttöön siirto. Nämä työvaiheet on jaettu iteraatioihin, joilla kuvataan työvaiheen kestoa ajallisesti. Kulloisenkin työvaiheen aikana suoritetaan tehtäviä joita voisi kutsua sisällön käyttäytymistä ohjaaviksi säännöiksi. Kuviossa 2 on esiteltynä mitä

tehdään milloinkin ja kuinka paljon painoarvoa/työmääriä millekin tehtävälle laitetaan missäkin vaiheessa. (Cottrell, W. 2004.)

3.1.3 Scrum

Scrum perustuu empiiriseen prosessinhallinta malliin. Työn suunnittelu ja toteutus jaetaan pienempiin työvaiheisiin (sprintit) ja jokainen näistä työvaiheista testataan. Tulokset ja onnistumiset haetaan testausten kautta. Kuten kuviossa 3 voi nähdä, projektin alussa kerätään yrityksen vaatimukset ja olosuhteet ja näistä vaatimuksista muodostetaan Product Backlog. Scrum tiimi priorisoi Product Backlogissa olevat vaatimukset (User Stories) tärkeysjärjestykseen ja aloittavat tämän pohjalta sprintin suunnittelun. Tämän ensimmäisen sprintin (Sprint 0) aikana priorisoidaan tehtävät ja jaetaan ne sprintteihin (Sprint 1 – X). Sprintin pituus on yleensä noin 4 viikkoa, mutta sen pituus voi vaihdella projektikohtaisesti. (Schwaber, K. & Beedle, M., 2002., s89)

Jokainen Sprint alkaa sprintin suunnittelulla, jossa päätetään mitä tehtäviä kyseisen sprintin aikana toteutetaan. Ryhmä avaa sprinttiin kerätyt käyttäjätarinat (User Story) pienemmiksi erotelluiksi tehtäviksi. Projektiryhmä kokoontuu sprintin aikana päivittäin pitämään daily scrumin ennen työn aloittamista. Daily Scrumissa käydään läpi mitä on tehty, mitä tullaan tekemään ja onko työlle ilmennyt esteitä ja miten mahdolliset esteet voidaan poistaa. Projektiryhmää ohjaa Scrum master, joka on vastuussa projektiryhmästä, työn etenemisestä ja työn tuloksista. Scrum masterin virka vastaaosittain projektipäällikköä. Sprintin lopulla tuotetut tulokset esitellään asiakkaalle, joka joko hyväksyy tulokset tai ehdottaa muutosta. Tästä alkaa jälleen uusi sprintti kierros, joka jälleen aloitetaan suunnittelulla ja työnjaolla. Lopuksi kun kaikki sprintit on valmiita, tulisi ohjelman olla valmis ja se esitellään ja toimitetaan asiakkaalle. (Schwaber, K. & Beedle, M., 2002., s89)

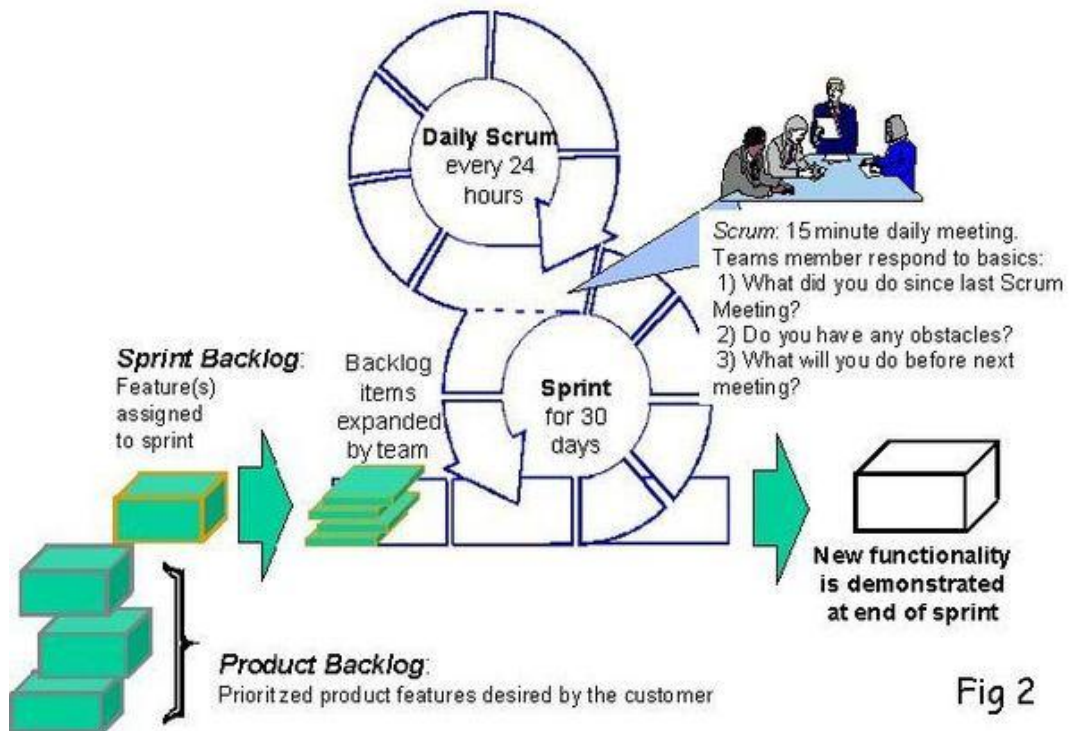


Fig 2

Kuvio 3. Scrumin vaiheet (AgilistaPM)

Yllättäviä ja ennalta arvaamattomia asioita tapahtuu lähes aina kehitysprojektin aikana. Scrum on siitä hyvä, että se antaa varaa ja myöten näille ennalta arvaamattomille muutoksille. Vaikka muutoksen tarve ilmaantuu projektin aikana, ei ole liian myöhäistä tehdä muutosta. Kuten Schwaber ja Beedle (2002) kirjassaan sanovat ”Scrum expects every process to be unexpected.”.

Scrumissa on ideana että projektiryhmä on itseohjautuva ja projektiryhmällä on vapaus tutkia ja selvittää asioita, mutta tämän lisäksi heiltä odotetaan korkeatasoista ja laadukasta jälkeä/tulosta. Tiimi työskentelee yhtenäisesti ja läheisessä vuorovaikutuksessa keskenään, tieto kulkee ja jakautuu ryhmän sisällä.

Jotta Scrum toimisi ihanteellisesti, on kommunikaation oltava hyvä ja avoin asiakkaan suuntaan. Scrumissa työryhmä on yleensä suoraan kontaktissa asiakkaaseen (ei erillistä välittäjä). Tämä vähentää kommunikaatio ongelmia, sekä ei toivottuja tuloksia.

Koska järjestelmävaatimukset ovat tuskin koskaan määritelty kokonaan, ohjelmistokehittämiseen kuuluu myös selvitys- ja tutkimustyö. (Schwaber, K. & Beedle, M., 2002, s107.)

Asiakkaalta vaaditaan aktiivista osallistumista ja jos asiakas ei osallistu tarpeeksi aktiivisesti projektin testaukseen tai palavereihin, voi sillä olla suuria vaikutuksia tuloksen laatuun. Asiakkaan on tärkeää testata toimiva ohjelma joka sprintin katselmoinnissa (Sprint Review). Tällä varmistutaan siitä, että ryhmä tuottaa sovittua tulosta. Scrumissa riskinä on, että kaikkia toiminnallisuuksia ei saada toteutettua. Tähän on ratkaisuna tehtävien priorisointi, korkeamman prioriteetin omaavat toiminnallisuudet toimitetaan julkaisuun, ja alhaisemman prioriteetin jos on aikaa toteuttaa. (Schwaber, K. & Beedle, M., 2002, s109.)

3.2 Kuvauskäytännöt

Oliokeskeisen ohjelmistokehittämisen yleisin kuvaustekniikka on UML (Unified Modeling Language). UML on kokoelma erilaisia käsitteitä ja kaavioita. Mallinnuskaavioilla kuvataan järjestelmän rakennetta ja käyttäytymistä. Yleisimmin käytettyjä UML-kuvaustekniikoita ovat käyttötapauskaavio, luokka- ja oliokaavio, yhteistyökaavio. Tässäkin projektissa on käytetty UML:n mukaisia kuvauksia mm suunnittelukuvasto. (Sommerville, I, 2007, s11, s153.)

4 Projektin tavoitteet

Projekti on lähtöisin Suomen Mäyräkoiraliiton hallituksen päätöksestä tuottaa uusi versio tietokannasta. Myös tallentajille ja jalostuksen tueksi toivottiin toimivampaa käyttöliittymää. Projekti on erillinen osa-alue suuremmasta maykkyNet -projektista. Työn rajaus ei ole kuitenkaan riippuvainen muusta projektista.

4.1 Valitut työtavat ja -välineet

Projektissa valittiin käytettäväksi iteratiivista lähestymistapaa, joka sopii pienille projektiryhmille ja kiireiseen aikatauluun. Tämä työtapo sopii paremmin myös projektille, joka elää kehityksen mukana. Nopean kehittämistahdin vuoksi määrittäyskuvastoa ei tehty vaan työstä on laadittu Product Backlog, jossa vaatimukset on esitetty käyttäjätarinoina (User Stories). Product Backlog on järjestetty lista kaikesta, mitä tuotteessa saatetaan tarvita, sekä ainoa lähde tuotteeseen toteutettaville vaatimuksille ja muutoksille (Sininen Meteoriiitti, 2011). Tuotteen kehittäminen ja hyväksyminen tapahtuu järjestelmän testa-

usten kautta. Tuotteen testaus on tarkemmin esiteltynä luvussa 4.6 Testaus ja tulosten seuranta.

Suomen Mäyräkoiraliitto on yhdistys eikä sillä ole käytössään erillistä budjettia, sillä tapahtuva työ tehdään talkootyönä. Tämän vuoksi projektissa suositetaan avoimen lähdekoodin ohjelmia, mikä on vaikuttanut moniin valintoihin työvälineiden suhteen. Avoimen lähdekoodin ohjelmat ovat vapaasti ladattavissa Internetistä ja niiden lähdekoodia voi muuttaa omaan tarkoitukseensa sopivaksi.

Ohjelmointikielenä projektissa käytetään PHP:ta, joka soveltuu selainpohjaisten sovellusten toteuttamiseen. Työssä käytetään CodeIgniter framework:iä, joka on MVC-mallin(Model-View-Controller) mukainen ohjelmakehys ja kuuluu avoimen lähdekoodin ohjelmiin. Kontrolleri hoitaa tietojen välityksen tietoluokkien ja näkymien välillä. Tähän toteutustapaan päädyttiin toimittajan toivomuksesta ja taitojen pohjalta. Vaihtoehtona olisi ollut käyttää CakePHP framework:iä, jolla on rakennettu SML:n oma erillinen maykkyNet – projekti. CakePHP frameworkin käytön opettelu ja siihen tutustuminen todettiin kuitenkin tähän projektiin liian aikaa vieväksi, eikä sitä siksi otettu käyttöön. CodeIgniter:in valintaan frameworkiksi päädyttiin siksi, että se on kevyempirakenteinen, selkeä ja sopii paremmin pienemmälle sovellukselle. Lisäksi, sille löytyy hyvät käyttöohjeet CodeIgniterin kotisivuilta. PHP:n valinta ohjelmointikieleksi on toimeksiantajan määrittelemä. Muita mahdollisia vaihtoehtoja olisivat olleet esimerkiksi ASP.Net, Java, JavaScript tai Perl, jotka myös soveltuvat web sovelluksen kehittämiseen.

Työvälineenä käytetään Eclipsen Helios versiota, joka soveltuu PHP sovelluskehitykseen. Eclipsen valintaan työvälineeksi päädyttiin toimittajan aiemman käyttökokemuksen myötä ja uuden ohjelmatyökalun opetteluun olisi kulunut liikaa aikaa projektilta.

Toimeksiantajan toiveena oli, että projektissa käytetään tietokantana MySQL, ja tästä toiveesta päädyttiin valintaan käyttää kehitysalustana XAMPP Lite Windowsille. Se on avoimen lähdekoodin ohjelma ja on ilmaiseksi saatavilla ja ladattavissa Internetistä.

XAMPP sisältää seuraavat komponentit:

- Apache 2.4.3 serveri

- MySQL 5.5.27 tietokanta
- PHP 5.4.7
- phpMyAdmin 3.5.2.2 tietokannan hallintatyökalu

4.2 Ketterän kehityksen menetelmät

Projektin aikana voi ilmaantua muutoksia, jotka on projektin kannalta tärkeä mahdollistaa, sillä kaikkea ei voida etukäteen tietää ja suunnitella. Ketterät menetelmät ovat joustavia muutoksille, sillä projektia työtetään pala kerrallaan. Ongelmat voidaan löytää ajoissa ja niihin voidaan puuttua ennen kuin on aiheutunut mitään suurempaa vahinkoa. Mahdolliset muutokset saadaan nopeammin toteutettua, eikä koko projektia tarvitse aina alusta asti ryhtyä muuttamaan. Näin resurssit pystytään ohjaamaan sinne, missä niitä eniten tarvitaan ja missä niistä on eniten hyötyä, sekä minimoitua turhaa työtä. Ketterät menetelmät on suunniteltu tukemaan ohjelmistokehitystä nimenomaan muuttuvassa ympäristössä, ja ne mahdollistavat nopean reagoimisen parempaan tietoon ja ymmärrykseen vaatimuksista. Ketterän kehityksen projekteissa tehtävät ja toiminnallisuudet priorisoidaan ja tärkeimmät ja kriittisimmät ominaisuudet toteutetaan ensimmäisinä. Tämä toimintatapa auttaa myös ongelmien havaitsemisessa. (Lukumaa, T., Reaktor 2011.)

Asiakas on yleensä aktiivisesti mukana projektissa, jolloin tieto asiakkaan ja toteutustien välillä kulkee selkeämmin ja työn tilanne pysyy molemmille ajantasalla. Ketterän kehityksen projekteilla ei yleensä ole varsinaista projektipäällikköä, vaan projektiryhmässä toimii yleensä Scrum Master, joka on ryhmässä vastuussa ja ohjaa sen työskentelyä. (Lukumaa, T., Reaktor 2011.)

Scrumia on käytetty lähtökohtana tämän projektin kehittämisessä ja vaikka sitä ei ole suoraan sellaisenaan käytetty, on sen menetelmiä hyödynnetty projektin kehittämisessä. Koska projektiryhmä on pieni, sopii ketterät menetelmät hyvin projektin kehittämiseen. Projektin kulku on jaettu pariin työjaksoon, joiden aikana suunnitellaan ja toteutetaan työ. Tämä vastaa sprinttejä. Product Backlogilla on hoidettu kommunikaatio suuren määrittämyskuvaston sijasta, kuten scrumissakin. Toimittaja testaa jokaisen toteutetun osan toimivuuden, joka vastaa Sprint Review -vaihetta. Asiakkaan suorittamia testauksia on

kuitenkin vähemmän kuin ketterän kehityksen mallissa olisi suositeltavaa. Tässä projektissa ei myöskään asiakas ole ollut niin aktiivisesti mukana vaan toimittaja on saanut työstää toteutusta omassa aikataulussa.

4.3 Projektin tehtävä

Projektin tehtävänä oli tuottaa SML:n tallentajille mäyräkoirien jalostustarkastuksessa käytettävä itsenäisesti toimiva lokaali järjestelmä, jolla tallentaja voi helposti hakea koiran tiedot ja kirjata jalostustarkastuksen aikana kerätyt tiedot tietokantaan. Kyseinen sovellus toimii lokaalisti tarkastajan tietokoneella. Tämä projekti vastaa vain lokaaliver-siosta ja lokaalin kannan suunnittelusta, ei tietojen synkronoinnista isoon kantaan tai tietojen migratoinnista vanhasta kannasta uuteen. Lisäksi tuotetaan ohjeet käyttöliittymän käyttöä, ylläpitoa sekä mahdollista jatkokehitystä varten.

4.4 Tavoitteet ja lopputulokset

Projektin päätavoitteena on tuottaa jalostustarkastuksiin lokaalisovellus jalostustoimikunnan tarpeisiin. Tuloksena projekti tuottaa käyttöliittymän jalostusrekisterin tallentajille ja lokaalin tietokannan tietojen tallennusta varten. Tavoitteena on selkeyttää jalostustarkastusraportin ulkoasua ja luettavuutta.

Työssä tulee huomioida ohjelman ylläpidettävyys sekä mahdollinen jatkokehitys. Ohjelma on hyvin ylläpidettävä silloin, kun siihen on helppo tehdä tarpeen sattuessa muutoksia ja päivityksiä. Projektin tulokselle asetetut vaatimukset on eriteltynä Product Backlogissa.

Oppimistavoitteena oli tuottaa PHP:llä toimiva, siisti ja käyttäjäystävällinen käyttöliittymä. Tämä mahdollistaa sen, että käyttäjä osaa käyttää ohjelmaa ilman erillistä käyttöohjetta, ohjelma on ulkoasullisesti siisti ja helppolukuinen ja mukana on vain käyttäjän tarvitsemat toiminnot. Tavoitteena oli myös oppia keskustelemaan ja toimimaan asiakkaan kanssa yhteistyössä ja toteuttaa asiakkaan toiveita mahdollisimman suureltaosin täyttävä työ.

4.5 Valmis ohjelma – kenelle tulee

Valmis työ tulisi käyttöön kaikille SML:n tallentajille ja tuloksen on tarkoitus helpottaa ja nopeuttaa heidän työtään, sekä täyttää nykyiset vaatimukset. Työn tulos olisi hyödynnettävissä myös maykkyNet-projektin muihin osakokonaisuuksiin. SML toimii tal-kootyöllä ja edullinen toteutustapa on toivottua. Valmis työ esitetään projektin omistajalle (Suomen Mäyräkoiraliitto) ja sidosryhmille.

5 Jalostustarkastus-projekti

Tässä luvussa kerrotaan projektin vaiheet, kuinka on menetelty ja mitä on tehty. Taustat ja lähtötiedot on kuvattu luvussa 4.

5.1 Projektin aloitus

Projektista on laadittu projektisuunnitelma, joka hyväksyttiin projektin aloituskokouksessa muutoksin.

5.2 Määrittäminen

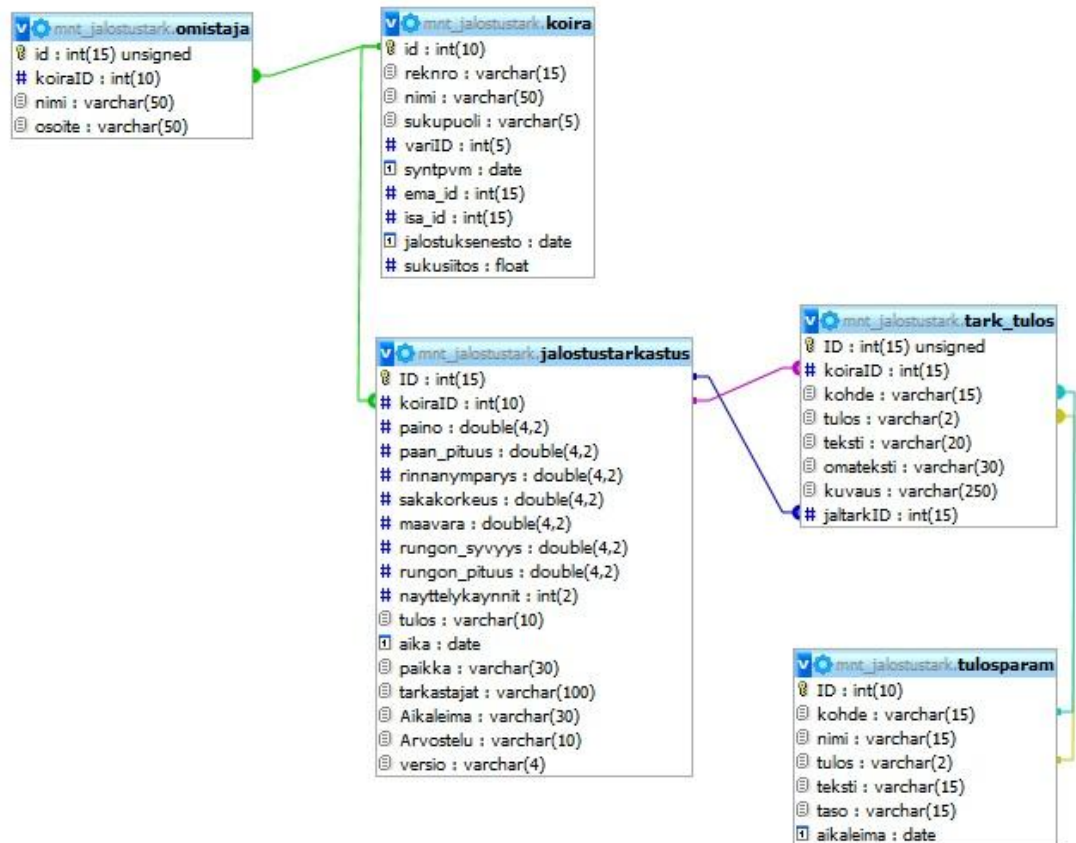
Projektin alussa sovittiin, ettei erillistä määrittämissuunnitelmaa toteuteta, vaan on laadittu keuhityksen mallin pohjalta Product Backlog, jossa vaatimukset on esitelty käyttäjätarinaoina. Käyttäjätarinat ovat toimeksiantajan projektin alussa esittämiä toiminnallisia vaatimuksia, jotka on priorisoitu niiden tärkeyden mukaan. Product Backlogista nähdään toteutetut ja toteutumattomat vaatimukset ja niille asetetut prioriteetit.

Projektissa olisi ollut mahdollista tehdä kuvaukset käyttötapauksista, joka on UML mallinnuksen mukainen vaihtoehto. Käyttötapaukset kuvataan yleensä määrittämissuunnitelmassa. Käyttötapauksia ei ole kuvattu tässä projektissa, sillä niitä ei ole tarvittu.

5.3 Suunnittelu

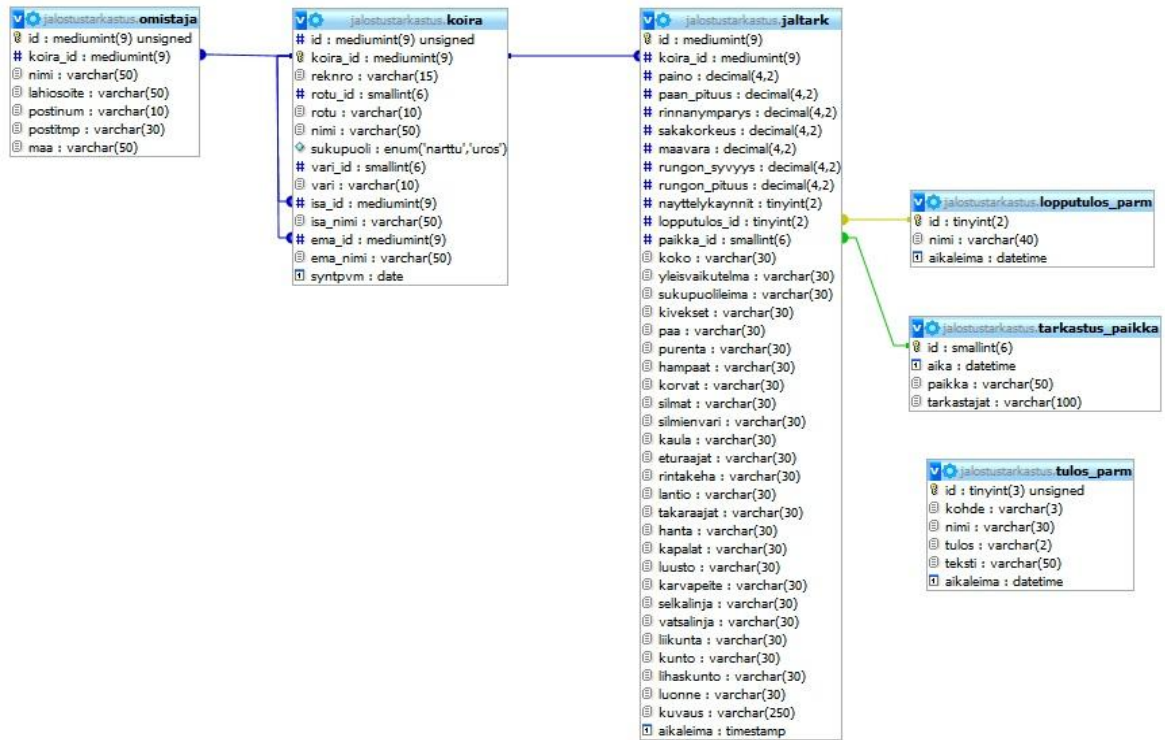
Suunnittelukuvastossa on käytetty UML mallinnusta ja se on laadittu, jotta asiakas saa raportoidun kuvauksen työn suunnittelusta ja sen sisältörakenteesta. Suunnittelukuvasto on laadittu myös toteuttajan tueksi ohjaamaan työn toteutusta, sekä jatkokehittäjille kuvaukseksi järjestelmästä. Sen tärkeys on tässä projektissa korostunut, sillä tarkempaa määrittämissuunnitelmaa ei ole laadittu.

Luokkakaaviosta tehtiin kaksi versiota. Versiot on esiteltynä kuvioissa 4 ja 5. Luokkakaavion ensimmäinen versio todettiin liian puutteelliseksi ja epäkäytännölliseksi.



Kuvio 4. Luokkakaavion 1. versio

Luokkien välillä ei haluttu säilyttää monisidoksia suhteita ja luokkien sisällöt olivat osittain puuttellisia. Koska tietokanta haluttiin pitää mahdollisimman kevyenä, mietittiin kenttien tyyppiarvoja ja pituuksia uudelleen. Huomattiin, että joidenkin kenttien arvot olivat turhan suuria ja että pienemmätkin arvot riittävät. Suunniteltiin hieman erilaisella rakenteella uusi tietokanta, joka on kuvattuna kuviossa 5.



Kuvio 5. Luokkakaavion 2. versio.

Uuteen luokkakaavioon luokkien väliset suhteet korjattiin toivotumman laiseksi ja luotiin muutama uusi luokka lisää. Lisäksi joitakin luokkia poistettiin tai niitä yhdistettiin keskenään.

5.4 Toteutus

Projektin toteutusvaiheessa suunnitelmien pohjalta toteutettiin uusi järjestelmä. Toteutusta ei jaettu pienempiin osakokonaisuuksiin vaan koko järjestelmä toteutettiin yhdessä työvaiheessa. Työn toteutuksessa suositettiin avoimenlähdekoodin ohjelmia ja muita mahdollisia ilmaisohjelmia, sillä projektille ei ollut varattuna budjettia. Lokaali tietokanta toteutettiin, ennen käyttöliittymää. Koska tietokannan osalta kaikkea ei oltu osattu ottaa huomioon työn suunnitteluvaiheessa, muutoksia tietokantaan ilmeni myös toteutusvaiheessa ja näin ollen suunnittelu ja toteutus vaihe menivät jossain vaiheessa lomit-
tain.

5.5 Testaus ja tulosten seuranta

Ohjelmistotestaus voidaan suorittaa testitapauksilla tai tutkivalla testauksella. Testitapauksilla testattaessa sovellus testataan tekemällä testauslomakkeella esitetyt asiat, jotka ohjaavat testaajaa. Tutkivassa testauksessa testaajalla ei ole suoritettavaa tehtävälistaa vaan hän suorittaa testauksen oman asiantuntemuksensa pohjalta ja antaa palautteen sovelluksen toiminnallisuudesta ja hyödyllisyydestä. Testauksella mahdollistetaan puutteiden, virheiden ja epäkäytännöllisten asioiden löytyminen ja niiden korjaaminen ajoissa. (S. Korhonen, 2011.)

Toimittaja testaa sovelluksen osien toimivuuden ja tekniset ominaisuudet jo kehitysvaiheessa. Lopullisen tuotteen testaa kuitenkin toimeksiantaja. Jalostustarkastuksen tallentaja testaa tuotteen toimivuuden ja käytettävyyden testitapauksilla, sekä suorittaa myös tutkivan testauksen. Näin varmistetaan, että sovellus toimii oikein, mutta täyttää myös jalostustarkastuksen edellyttämät vaatimukset. Testauksessa on käytetty testauslomaketta (Liite 4 Testauslomake), joka on tehty testaajille avuksi ja ohjeistukseksi.

Testitapaukset tehtäisiin normaalisti jokaisesta käyttötapauksesta, mutta koska tässä projektissa niitä ei ole tehty, poimitaan testitapaukset käyttäjätarinoista ja käyttöliittymäkuvauksista. Käyttäjätarinat on lueteltuna Product Backlogissa ja käyttöliittymäkuvaukset Suunnittelukuvastossa.

5.6 Projektin päättäminen

Työstä on kirjoitettu loppuraportti, jossa esitetään saavutetut tulokset ja kokemukset, työn edistyminen, sekä on esitelty ehdotukset mahdollisiksi jatkotoimenpiteiksi. Projektin päätöskokous pidetään marraskuussa 2012 johon osallistuu koko projektiorganisaatio.

6 Tulosten esittely

Työn tuloksena tavoiteltiin toimivaa selainpohjaista lokaalia järjestelmää, joka täyttää asiakkaan esittämät vaatimukset. Ohjelma täyttää tärkeimmän toiminnallisuusvaatimuksen, joka on jalostustarkastuksen suorittaminen. Tallentaja voi kirjata koiralle jalostustarkastustiedot ja tallentaa ne lokaaliin tietokantaan, josta ne myöhemmin siirretään maykkynet-kantaan. Tarkastuksen saavien koirien tiedot täytyy kopioida maykkynet-kannasta jalostustarkastus-kantaan import-export tyylisesti, sillä automaattista toiminnallisuutta ei tälle vielä rakennettu. Jalostustarkastus-kantaan poimitaan vain tarkastuksen osalta tarpeelliset tiedot koirasta ja omistajasta. Käyttäjä voi jalostustarkastuslomaketta täyttäessään hakea koiran ja omistajan tiedot lomakkeelle lokaalista tietokannasta koiran rekisterinumeron perusteella. Jalostustarkastustietojen syöttäminen tapahtuu kolmessa eri vaiheessa: syötetään koiran ja omistajan tiedot, syötetään koiran mitat, syötetään tulos. Kun jalostustarkastus on suoritettu, saa käyttäjä tulostusnäkyvän tarkastuksen tiedoista allekirjoitus viivoilla.

Kaikkia asiakkaan esittämiä toiveita ei projektissa saatu toteutettua. Uuden koiran tietojen lisääminen jäi toteuttamatta, koska lokaalissa jalostustarkastus-kannassa on koirasta suppeammat tiedot kuin itse maykkynet-kannassa. Tulokset vastaavat kuitenkin suurimmalta osin tavoitteita.

7 Pohdinta

Tässä luvussa vertaillaan työn toteutumista tehtyjen valintojen osalta, tarkastellaan oppimisprosessia ja esitellään jatkokehitysehdotukset.

7.1 Tulokset

Ketterän kehityksen tyyli sopi tähän projektiin, sillä työn sisältö muuttui pariin otteeseen projektin aikana. Muutokset ja rajaukset saatiin tehtyä ja projektia jatkettua. Käytettiin CodeIgniter-frameworkkiä, jossa oli tiettyjä funktioita valmiiksi tehtynä, kuten tietokantayhteyden muodostaminen ja kyselyn suorittaminen, jotka helpottivat ja

siisteyttivät koodin rakentamista. Framework auttoi myös joidenkin toimintojen toteutamisessa, jotka muuten olisivat osoittautuneet ongelmallisiksi.

Tuloslomakkeen asettelun toteutus ei onnistunut halutulla ja suunnitellulla tavalla. Parempi ja siistimpi vaihtoehto olisi ollut check box -rakenteen sijaan syöttökentät, joihin kohteiden tulokset olisi syötetty yksitellen ja tuloksena muodostaneet listauksen. Asiakas oli kuitenkin tyytyväinen check box – rakenteeseen, joten se päätettiin toteuttaa sillä tavalla.

Uuden koiran tietojen lisäämsien suhteen ei ollut tarvittavaa osaamista kannan rakenteen suunnitteluun. Asiaa yksinkertaistaisi jos olisi yksi tietokanta, johon tallennetaan ja josta haetaan, ei erikseen lokaalia tietokantaa.

7.2 Kustannusarviot ja työajanseuranta

Työtuntien perusteella oli tarkoitus laskea työstä tekninen budjetti, jonka toteutumista seurattaisiin ja asiakas saisi tuloksena arvion mahdollisista koituneista työkustannuksista. Tämä rajattiin lopullisesta toteutuksesta kuitenkin pois, sillä työtunnit on eroteltuna ja mahdolliset työkustannukset ovat laskettavissa taulusta Toteumaseuranta, joka löytyy liitteessä 3 Aikataulu toteuma.

7.3 Oman oppimisen arviointi

Projektin aikana kehityin valtavasti PHP tuntemuksen osalta. Opin lisää uusia ja erilaisia tekniikoita rakentaa selainpohjainen sovellus. PHP:llä en ollut aiemmin käyttänyt MVC-mallia, mutta tähän sain apua hieman kertyneistä kokemuksista Java maailmasta.

Projektin etenemisen ja prosessin kulun suhteen osaaminen oli odotettua heikompi. Prosessin vaiheet olisi kuulunut olla selkeitä jo tässä vaiheessa ja etenemisen ei olisi kuulunut olla niin hidasta, kuin se tässä projektissa oli.

Suurin haaste oli lokaalin tietokannan suunnittelu jalostustarkastuksen tallennusta varten. Osaamiseni kantojen suunnittelussa on aikaisemminkin ollut heikohkoa, ja tässä projektissa huonojen versioiden ja ideoiden jälkeen pääsin kohtalaiseen versioon. Ei

sitä muistanutkaan kuinka aikaa ja suurta mietintää vaativaa työtä pienenkin tietokannan suunnittelemiseen on.

7.4 Jatkokehitysehdotukset

Jatkokehityksenä ehdotetaan jalostustarkastus-tietokannan kehittämistä, sekä ohjelman siirtämistä palvelimelle ja siirtymään pois lokaalitoteutuksesta. Datan liikuttaminen eri tietokantojen välillä aiheuttaa nyt lisätyötä ja mahdollisia riskejä datan eheyden säilyttämisessä. Virheiden käsittelyä voisi viedä tarkemmalle tasolle.

Lisäksi suosittelaa työstä poisrajattujen ominaisuuksien käyttöönottoa: käyttäjän tunnistautuminen, koirien kopioiminen maykkynetistä lokaaliin-kantaan ja jalostustarkastusien siirto lokaalista kannasta maykkynet-kantaan.

Lähteet

AgilistaPM. How To Implement Scrum in 10 Easy Steps. Luettavissa:

<http://www.agilistapm.com/how-to-implement-scrum-in-10-easy-steps/>. Luettu 20.11.2012.

Apache friends 2012. XAMPP for Windows. Luettavissa:

<http://www.apachefriends.org/en/xampp-windows.html>. Luettu: 9.10.2012

Cotrell, W., Process and Portfolio Management Evangelist, IBM 2004. Standards, compliance, and Rational Unified Process, Part I: Integrating RUP and the PMBOK.

Luettavissa: <http://www.ibm.com/developerworks/rational/library/4763.html>. Luettu 20.11.2012

Haikala, I., Märijärvi, J. 2004. Ohjelmistotuotanto. 10. uudistettu painos. Talentum. Helsinki.

Lukumaa, T., Reaktor 29.4.2011. Ketterät eli Agile-menetelmät. Luettavissa:

<http://reaktor.fi/osaaminen/agile/>. Luettu 14.10.2012

Schwaber, K., Beedle, M. 2002. Agile Software Development with Scrum. Pearson Education.

Sirén, S. 2011. Avainhallintasovelluksen käyttötapausten suunnittelu.UML, s11-15.

Amk-opinnäytetyö. Kemi-Tornion ammattikorkeakoulu. Tornio. Luettavissa:

https://publications.theseus.fi/bitstream/handle/10024/30010/Siren_Sami.pdf?sequence=1. Luettu: 13.11.2012

Sininen Meteriitti 10.8.2011. Ketteryys haltuun: Ketterän kehityksen yleiset periaatteet. Luettavissa:

<http://www.meteoriitti.com/fi-FI/tiedotteet/ajankohtaista/ketteryys-haltuun-ketteran-kehityksen-yleiset-periaatteet>. Luettu 2.10.2012

Sommerville, I. 2007. Software Engineering. 8. uudistettu painos. Pearson Education Limited.

Suomen Mäyräkoiraliitto SML Ry a. Mäyräkoiraliitto. Tietoa Mäyräkoiraliitosta. Luettavissa:

http://www.mayrakoiraliitto.fi/index.php?option=com_content&view=article&id=2&Itemid=185. Luettu 11.9.2012.

Suomen Mäyräkoiraliitto SML Ry b. Jalostus. Jalostustarkastus. Luettavissa:

http://www.mayrakoiraliitto.fi/index.php?option=com_content&view=article&id=33&Itemid=208. Luettu 11.9.2012.

Tietoviikko, Korhonen, S. 17.11.2011. Tutkiva ohjelmistotestaus säästää rahaa ja löytää yllättävämmänkin reiän. Luettavissa:

http://www.tietoviikko.fi/kaikki_uutiset/tutkiva+ohjelmistotestaus+saastaa+rahaa+ja+loytaa+yllattavammankin+reian/a724344. Luettu 4.11.2012.

Liitteet

Liite 1. Product Backlog

Story ID	Story name	Status	Sprint	Priority	Työmäärä	arvio	Comments
1	Käyttäjänä haluan hakea koiran tiedot maykkynet -tietokannasta ja ladata ne jalostustarkastus tietokantaan	Removed	1	3	20		Koirat joille jalostustarkastus tehdään, noudetaan maykkynet-tietokannasta ja talletetaan väliaikaisesti jalostustarkastus-tietokantaan
2	Käyttäjänä haluan tallentaa jalostustarkastuksen tiedot jalostustarkastus -tietokantaan	Done	1	2	10		Liittyy storyyn 3.
3	Käyttäjänä pystyn suorittamaan jalostustarkastuksen ja kirjata tulokset ohjelmalla	Done	1	1	60		tarkastuksen tallentaminen tietokantaan on kohdassa 2
4	Käyttäjänä saan tarkastuksesta tulostusnäytymän allekirjoitusviivoinla	Done	1	4	5		
5	Käyttäjänä pystyn siirtämään tiedot jalostustarkastus -tietokannasta maykkynet -tietokantaan	Removed	1	5	20		
6	Käyttäjänä voin halutessani lisätä uuden koiran tiedot järjestelmään	Planned	1	7	10		
7	SML:n teknisenä suunnittelijana minulla on järjestelmän ylläpitoon ohjeistava käyttöohje	Planned	1	8	5		
8	Käyttäjänä minulla on käytössäni käyttöohjeet, jotka opastavat ohjelman käytössä	Planned	1	9	5		
9	Teknisenä suunnittelijana haluan että koodi on selkeää ja kommentoitua, jotta ohjelman jatkokehittäminen olisi helpompaa	Ongoing	1	6	3		Liittyy osaltaan jo storyihin 1, 2 ja 3, sillä kommentteja tehdään koodia kirjoittaessa, sekä täydennetään tarvittaessa koodin ollessa jo valmista
10	Käyttäjänä voin kirjautua sisään järjestelmään saaduilla käyttäjä-	Removed	1	10	2		

tunnuksilla

Liite 2. Suunnittelukuvasto

Suunnittelukuvasto

HAAGA-HELIA amk

Jalostustarkastus-projekti

Iris Erkkilä

Versio	0.5	Ehdotus
Tekijät	Iris Erkkilä	23.11.2012
Tarkastanut	Iris Erkkilä	23.11.2012
Hyväksynyt	<hyväksyjän nimi>	<hyv. pvm>

MUUTOSHISTORIA

versionumero	pvm	muutos
0.1	31.7.2012	suunnittelukuvaston ensimmäinen versio
0.2	16.10.2012	suunnittelukuvaston toinenversio
0.3	21.10.2012	suunnittelukuvaston kolmasversio
0.4	30.10.2012	suunnittelukuvaston neljäs versio
0.5	23.11.2012	suunnittelukuvaston viimeistely

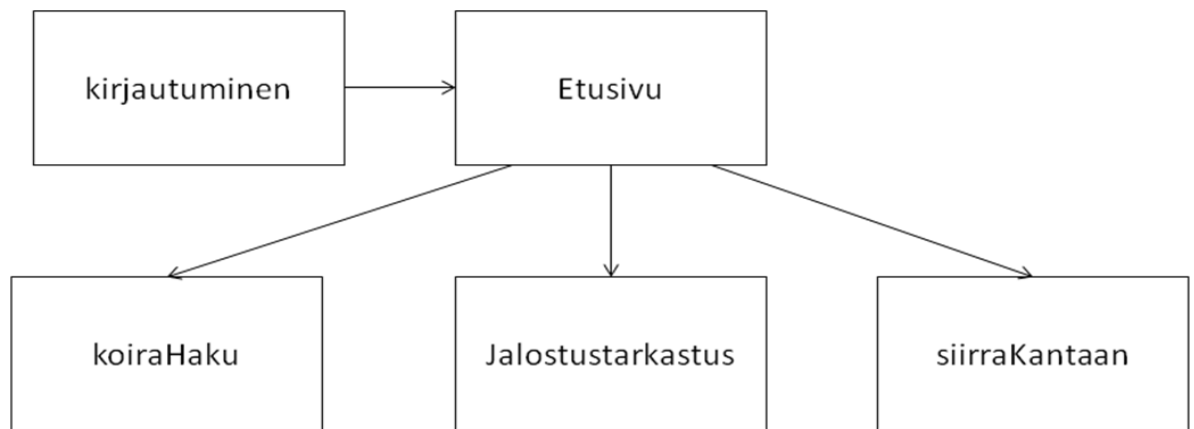
SISÄLTÖ

1. Johdanto
2. Käyttöliittymät
3. Luokkakaavio
4. Tietokanta
5. Arkkitehtuurimalli
6. Yhteistyökaaviot
7. Viestiyhteykskaaviot
8. Paketit

1. Johdanto

Tässä dokumentissa on kuvattu Jalostustarkastus-projektin, Suomen Mäyräkoiraliitolle suunnittelema sovellus. Tämä dokumentti sisältää käyttöliittymäsuunnitelman, luokka-kaavion, tietokantasuunnitelman sekä yhteistyökaavion, viestiyhteyksikaavion ja kuvauksen paketeista.

2. Käyttöliittymät



Kuva 1. Käyttöliittymät

Tässä luvussa kuvataan Jalostustarkastus-sovelluksen käyttöliittymät. Kuvassa 1 näkyy sovelluksen käyttöliittymien rakenne. Sovelluksen käyttöliittymät ovat kirjautuminen, etusivu, koiraHaku, jalostustarkastus ja siirraKantaan.

Tervetuloa Jalostustarkastuksen etusivulle!

[Tee tarkastus](#)

[Hae koirat](#)

[Tee tarkastus](#)

Kuva 3. Etusivu

Käyttäjä ohjataan suoraan sovelluksen etusivulle. Etusivulla käyttäjällä on valittavana seuraavat toiminnot (linkit): uusi jalostustarkastus, koiran haku maykkyNet kannasta ja koirien siirtäminen maykkyNet kantaan.

Koiran tiedot

Reknro

Nimi

Sukupuoli

Väri

Isä

Emä

Syntynyt

Omistaja

Omistaja

Osoite

Maa

Kuva 4. Koiran ja omistajan tietojen syöttö -lomake

Etusivulta ”Tee tarkastus” –linkistä käyttäjä saapuu Jalostustarkastus-sivulle, jossa otsikon alla ovat tekstikentät koiran tietojen täyttämistä varten: reknro, koiran nimi, sukupuoli, väri, isa, emä, syntynyt(kalenterivalinta), omistajan nimi, omistajan osoite, maa.

Käyttäjä voi hakea koiran ja omistajan perustiedot kannasta koiran rekisterinumeron perusteella. Jos käyttäjä syöttää kentät itse, tarkistetaan ne kannasta. Jos koiraa ei löydy kannasta, ohjelma luo tiedot kantaan.

Peruuta-painikkeesta käyttäjä ohjataan järjestelmän etusivulle.

Kuvan 4 näkymästä käyttäjä saapuu jalostustarkastuksen seuraavaan vaiheeseen, jossa on kentät koiran mitattavien ominaisuuksien täyttämiseen: paino, pään pituus, rinnan ympäryys, säkäkorkeus, maavara, rungon syvyys(laskennallinen kenttä, käyttäjällä vain lukuoikeus kenttään), rungon pituus, näyttelykäynnit(valintalista). (kuva 5). Peruuta-painike ohjaa käyttäjän takaisin etusivulle. Edellinen-painikkeesta käyttäjä ohjataan takaisin edelliseen näkymään (kuva 4).

Mitat

Paino

Pään pituus

Rinnanympärys

Säkäkorkeus

Maavara

Rungon syvyys

Rungon pituus

Näyttelykäynnit

-- Valitse --

Edellinen

Jatka

Peruuta

Kuva 5. Koiran mittojen syöttäminen

Kuvan 5 näkymästä Jatka-painikkeesta käyttäjä saapuu jalostustarkastuksen kolmanteen vaiheeseen, jossa on kentät tarkastuksen tulosten täyttämiseksi: koko, yleisvaikutelma, sukupuolileima, kivekset (jos sukupuoli on uros), pää, purenta, hampaat, korvat, silmät, silmienväri, karvanväri, pigmentti, kaula, eturaajat, rintakehä, lantio, takaraajat, häntä, käpälät, luusto, karvapeite, selkälinja, vatsalinja, liikunta, kunto, lihaskunto, luonne, kuvaus. Myös tarkastuspaikan tietojen syöttökentät ovat tässä näkymässä: aika, paikka, tarkastajat, tulos.

Tarkastustulos

Kohde	Tulos
Koko	<input type="checkbox"/> 1A <input type="checkbox"/> 1B <input type="checkbox"/> 1C <input type="checkbox"/> 2A <input type="checkbox"/> 2B <input type="checkbox"/> 3A <input type="checkbox"/> 3B
Yleisvaikutelma	<input type="checkbox"/> 1A <input type="checkbox"/> 2A <input type="checkbox"/> 2B <input type="checkbox"/> 2C <input type="checkbox"/> 2D <input type="checkbox"/> 2E <input type="checkbox"/> 3A <input type="checkbox"/> 3B <input type="checkbox"/> 3C <input type="checkbox"/> 3D
Sukupuolileima	<input type="checkbox"/> 1A <input type="checkbox"/> 2A <input type="checkbox"/> 3A
Kivekset	<input type="checkbox"/> 0A <input type="checkbox"/> 0B <input type="checkbox"/> 1A <input type="checkbox"/> 3A <input type="checkbox"/> 3B
Pää	<input type="checkbox"/> 1A <input type="checkbox"/> 2A <input type="checkbox"/> 2B <input type="checkbox"/> 2C <input type="checkbox"/> 2D <input type="checkbox"/> 2E <input type="checkbox"/> 2F <input type="checkbox"/> 2G <input type="checkbox"/> 2H <input type="checkbox"/> 3A <input type="checkbox"/> 3B <input type="checkbox"/> 3C
Purenta	<input type="checkbox"/> 0A <input type="checkbox"/> 0B <input type="checkbox"/> 0C <input type="checkbox"/> 0D <input type="checkbox"/> 1A <input type="checkbox"/> 1B <input type="checkbox"/> 1C <input type="checkbox"/> 2A <input type="checkbox"/> 2B <input type="checkbox"/> 3A
Hampaat	<input type="checkbox"/> 0A <input type="checkbox"/> 1A <input type="checkbox"/> 1B <input type="checkbox"/> 1C <input type="checkbox"/> 2A <input type="checkbox"/> 3A
Korvat	<input type="checkbox"/> 1A <input type="checkbox"/> 2A <input type="checkbox"/> 2B <input type="checkbox"/> 2C <input type="checkbox"/> 2D <input type="checkbox"/> 2E <input type="checkbox"/> 2F <input type="checkbox"/> 3A <input type="checkbox"/> 3B <input type="checkbox"/> 3C <input type="checkbox"/> 3D
Silmät	<input type="checkbox"/> 1A <input type="checkbox"/> 2A <input type="checkbox"/> 2B <input type="checkbox"/> 2C <input type="checkbox"/> 3A
Silmien väri	<input type="checkbox"/> 0A <input type="checkbox"/> 1A <input type="checkbox"/> 1B <input type="checkbox"/> 2A <input type="checkbox"/> 2B <input type="checkbox"/> 2C <input type="checkbox"/> 3A <input type="checkbox"/> 3B
Karvanväri	<input type="checkbox"/> 0A <input type="checkbox"/> 0B <input type="checkbox"/> 0C <input type="checkbox"/> 1A <input type="checkbox"/> 2A <input type="checkbox"/> 2B <input type="checkbox"/> 2C <input type="checkbox"/> 2D <input type="checkbox"/> 2E <input type="checkbox"/> 3A <input type="checkbox"/> 3B
Pigmentti	<input type="checkbox"/> 1A <input type="checkbox"/> 1B <input type="checkbox"/> 2A <input type="checkbox"/> 2B <input type="checkbox"/> 2C <input type="checkbox"/> 3A
Kaula	<input type="checkbox"/> 1A <input type="checkbox"/> 2A <input type="checkbox"/> 2B <input type="checkbox"/> 2C <input type="checkbox"/> 2D <input type="checkbox"/> 3A <input type="checkbox"/> 3B
Eturaajat	<input type="checkbox"/> 0A <input type="checkbox"/> 1A <input type="checkbox"/> 2A <input type="checkbox"/> 2B <input type="checkbox"/> 2C <input type="checkbox"/> 2D <input type="checkbox"/> 2E <input type="checkbox"/> 2F <input type="checkbox"/> 2G <input type="checkbox"/> 2H <input type="checkbox"/> 2I <input type="checkbox"/> 3A <input type="checkbox"/> 3B <input type="checkbox"/> 3C <input type="checkbox"/> 3D
Rintakehä	<input type="checkbox"/> 0A <input type="checkbox"/> 1A <input type="checkbox"/> 2A <input type="checkbox"/> 2B <input type="checkbox"/> 2C <input type="checkbox"/> 2D <input type="checkbox"/> 2E <input type="checkbox"/> 2F <input type="checkbox"/> 2G <input type="checkbox"/> 2H
Lantio	<input type="checkbox"/> 1A <input type="checkbox"/> 2A <input type="checkbox"/> 2B <input type="checkbox"/> 2C <input type="checkbox"/> 2D <input type="checkbox"/> 3A
Takaraajat	<input type="checkbox"/> 1A <input type="checkbox"/> 2A <input type="checkbox"/> 2B <input type="checkbox"/> 2C <input type="checkbox"/> 2D <input type="checkbox"/> 2E <input type="checkbox"/> 2F <input type="checkbox"/> 3A <input type="checkbox"/> 3B <input type="checkbox"/> 3C <input type="checkbox"/> 3D
Häntä	<input type="checkbox"/> 0A <input type="checkbox"/> 1A <input type="checkbox"/> 2A <input type="checkbox"/> 2B <input type="checkbox"/> 2C <input type="checkbox"/> 2D <input type="checkbox"/> 2E <input type="checkbox"/> 2F <input type="checkbox"/> 3A <input type="checkbox"/> 3B
Käpälät	<input type="checkbox"/> 1A <input type="checkbox"/> 2A <input type="checkbox"/> 2B <input type="checkbox"/> 2C <input type="checkbox"/> 3A <input type="checkbox"/> 3B
Luusto	<input type="checkbox"/> 1A <input type="checkbox"/> 1B <input type="checkbox"/> 2A <input type="checkbox"/> 3A
Karvapeite	<input type="checkbox"/> 1A <input type="checkbox"/> 2A <input type="checkbox"/> 3A
Selkalinja	<input type="checkbox"/> 1A <input type="checkbox"/> 2A <input type="checkbox"/> 2B <input type="checkbox"/> 2C <input type="checkbox"/> 2D <input type="checkbox"/> 2E <input type="checkbox"/> 3A <input type="checkbox"/> 3B <input type="checkbox"/> 3C
Vatsalinja	<input type="checkbox"/> 1A <input type="checkbox"/> 1B <input type="checkbox"/> 2A
Liikunta	<input type="checkbox"/> 1A <input type="checkbox"/> 2A <input type="checkbox"/> 2B <input type="checkbox"/> 2C <input type="checkbox"/> 2D <input type="checkbox"/> 2E <input type="checkbox"/> 2F <input type="checkbox"/> 3A <input type="checkbox"/> 3B
Kunto	<input type="checkbox"/> 1A <input type="checkbox"/> 2A <input type="checkbox"/> 2B <input type="checkbox"/> 3A <input type="checkbox"/> 3B
Lihaskunto	Luonne <input type="checkbox"/> 0A <input type="checkbox"/> 0B <input type="checkbox"/> 1A <input type="checkbox"/> 1B <input type="checkbox"/> 1C <input type="checkbox"/> 2A <input type="checkbox"/> 3A <input type="checkbox"/> 3B

Yleiskuvaus

Tarkastus

Aika

Paikka

Tarkastajat

Pisteet

Tulos

Edellinen

Jatka

Peruuta

Kuva 6. Tarkastuksen tulos

Lopuksi käyttäjälle avautuu tulostusnäkymä jalostustarkastuksesta.

Tuloste

Puppe	1234565-8
uros Syntynyt 2012-11-02	Väri: ruskea
Isä: Musti	1
Emä: Fluffy	2
Om: Marja Testinen	Osoite 4, 12345, Toimipaikka

Paino	5.5 kg
Rinnanympärys	58.3 cm
Rungon pituus	89 cm
Säkäkorkeus	18.3 cm
Rungon syvyys	11.70 cm
Maavara	6.6 cm
Pään pituus	23.2 cm

Koko	1A
Yleisvaikutelma	1A
Sukupuolileima	1A
Kivekset	0B
Pää	2A
Korvat	2B
Silmät	2A
Silmienväri	1A
Karvanväri	0A
Pigmentti	1A
Purenta	0B
Hampaat	1A

Tulosta ja Tallenna

Edellinen

Peruuta

Kuva 7. Tulostusnäkymä

Jos käyttäjä ei halukaan tehdä jalostustarkastusta, vaan painaa Peruuta-nappia, hänet ohjataan takaisin etusivulle ja tyhjennetään sessio.

Edellinen-painikkeista käyttäjä ohjataan aina edelliseen näkymään.

Jos järjestelmässä tapahtuu jokin virhe, käyttäjää pyydetään syöttämään tiedot uudelleen. Jos virhe toistuu, pyydetään käyttäjää yrittämään myöhemmin uudelleen.

Hae koira

Anna haettavan koiran id, rekisterinumero tai nimi:

Kuva 8. koiraHaku käyttöliittymä

Etusivulta käyttäjä saapuu koiraHaku-sivulle, jossa on syöttökenttä koiran hakemista varten maykkyNet kannasta. Käyttäjä syöttää koiran id:n, rekisterinumeron tai nimen ja painaa Hae-nappia. Tuloksena käyttäjä saa listauksen hakukriteerit täyttävistä koirista. Käyttäjä valitsee haluamansa koirat ja valitsee Lataa/Talleta. Ohjelma siirtää haetut koirat Jalostustarkastus tietokantaan.

Jos hakutuloksella ei löydy koiria, saa käyttäjä ilmoituksen ”Hakusanalla *bakusana* ei löytynyt yhtään koira”.

Palaa etusivulle -napista käyttäjä ohjataan takasisin etusivulle.

Jos järjestelmässä tai tietokantayhteyksissä tapahtuu jokin virhe, käyttäjä saa asiasta ilmoituksen.

Siirrä koirat

Kannasta löytyneet jalostustarkastukset

Yhteensä 0 tarkastusta

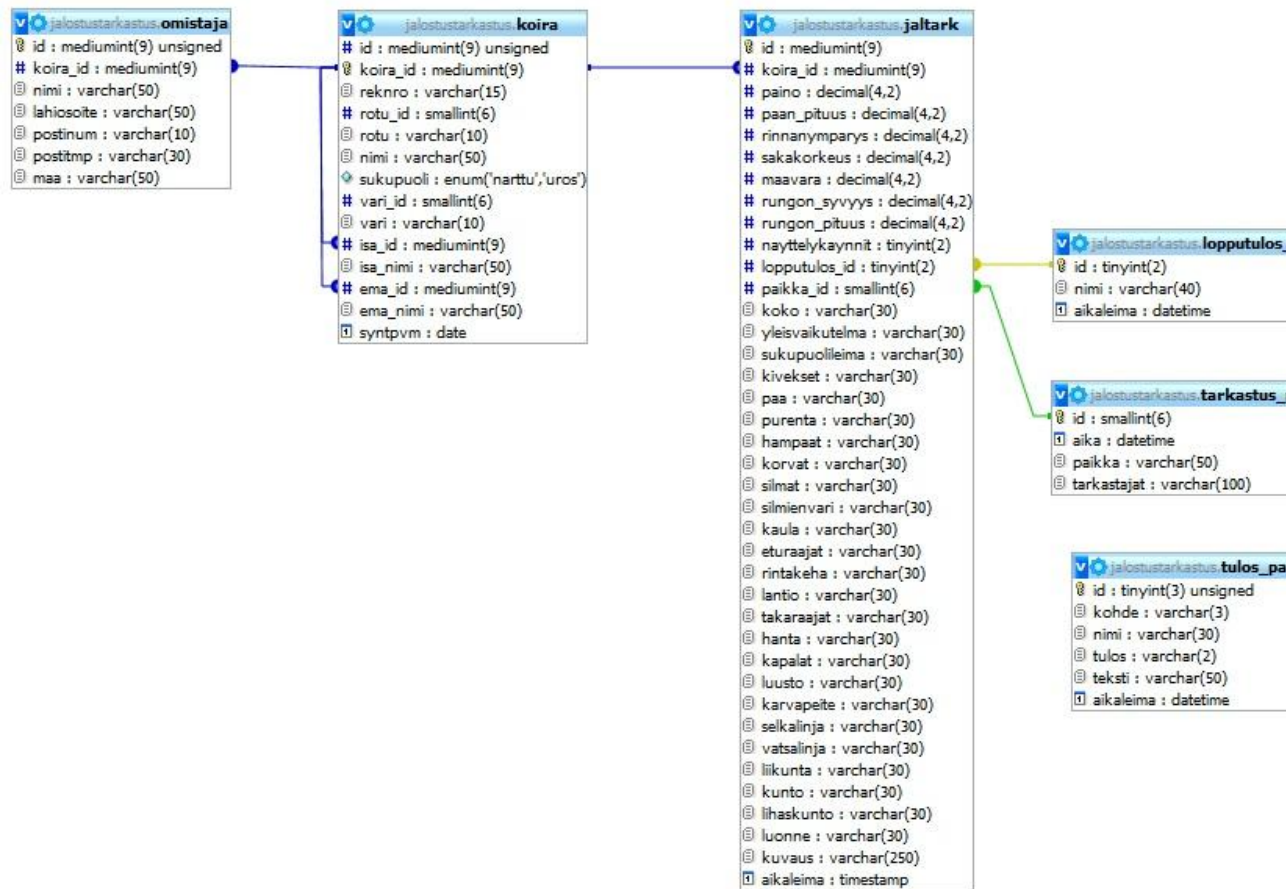
Kuva 9. siirraKantaan

Etusivulta käyttäjä saapuu siirraKantaan -sivulle, jossa hän voi siirtää jalostustarkastus tiedot maykkyNet -kantaan. Käyttäjä painaa nappia *Siirra koirat maykkyNettiin* ja järjestelmä siirtää toedot Jalostustarkastus-kannasta maykkyNet-kantaan. Käyttäjä saa ilmoituksen onnistuneesta siirrosta ”Tiedot on siirretty onnistuneesti”.

Palaa etusivulle-napista käyttäjä ohjataan takaisin etusivulle

Jos järjestelmässä tapahtuu virhe, pyydetään käyttäjää yrittämään uudelleen.

3. Luokkakaavio



Kuva 10. Luokkakaavio Jalostustarkastus sovelluksen osalta

4. Tietokanta

Ohessa on lokaalin tietokannan Jalostustarkastus luontilauseet. Kyseiset luontilauseet löytyvät myös projektin systeemityökansion alta sql tiedostona jalostustarkastus.sql. Sql tiedosto on mahdollista importata suoraan mysql tietokantaan, jolloin taulut muodostuvat kohdistettuun tietokantaan.

```
--  
-- Tietokanta: `jalostustarkastus`  
--  
  
-----  
  
--  
-- Rakenne taululle `jaltark`  
--  
  
CREATE TABLE IF NOT EXISTS `jaltark` (  
  `id` mediumint(9) NOT NULL AUTO_INCREMENT,  
  `koira_id` mediumint(9) NOT NULL,  
  `paino` varchar(5) NOT NULL,  
  `paan_pituus` varchar(5) NOT NULL,  
  `rinnanymparys` varchar(5) NOT NULL,  
  `sakakorkeus` varchar(5) NOT NULL,  
  `maavara` varchar(5) NOT NULL,  
  `runгон_syvyys` varchar(5) NOT NULL,  
  `runгон_pituus` varchar(6) NOT NULL,  
  `nayttelykaynnit` tinyint(2) NOT NULL,  
  `lopputullos_id` tinyint(2) NOT NULL,  
  `paikka_id` varchar(10) NOT NULL,  
  `koko` varchar(30) NOT NULL,  
  `yleisvaikutelma` varchar(30) NOT NULL,
```

```

`sukupuolileima` varchar(30) NOT NULL,
`kivekset` varchar(30) DEFAULT NULL COMMENT 'tyhjä silloin kun sukupuoli
narttu',
`paa` varchar(30) NOT NULL,
`purenta` varchar(30) NOT NULL,
`hampaat` varchar(30) NOT NULL,
`korvat` varchar(30) NOT NULL,
`silmat` varchar(30) NOT NULL,
`silmienvari` varchar(30) NOT NULL,
`kaula` varchar(30) NOT NULL,
`eturaajat` varchar(30) NOT NULL,
`rintakeha` varchar(30) NOT NULL,
`lantio` varchar(30) NOT NULL,
`takaraajat` varchar(30) NOT NULL,
`hanta` varchar(30) NOT NULL,
`kapalat` varchar(30) NOT NULL,
`luusto` varchar(30) NOT NULL,
`karvapeite` varchar(30) NOT NULL,
`selkalinja` varchar(30) NOT NULL,
`vatsalinja` varchar(30) NOT NULL,
`liikunta` varchar(30) NOT NULL,
`kunto` varchar(30) NOT NULL,
`lihaskunto` varchar(30) NOT NULL,
`luonne` varchar(30) NOT NULL,
`kuvaus` varchar(250) NOT NULL,
`aikaleima` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
PRIMARY KEY (`id`),
KEY `koira_id` (`koira_id`),
KEY `lopputulos_id` (`lopputulos_id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=11 ;

--
-- Vedos taulusta `jaltark`

```

--

--

-- Rakenne taululle `koira`

--

```
CREATE TABLE IF NOT EXISTS `koira` (  
  `id` mediumint(9) unsigned NOT NULL AUTO_INCREMENT,  
  `koira_id` mediumint(9) NOT NULL,  
  `reknro` varchar(15) NOT NULL,  
  `rotu_id` smallint(6) NOT NULL,  
  `rotu` varchar(10) NOT NULL,  
  `nimi` varchar(50) NOT NULL,  
  `sukupuoli` enum('narttu','uros') NOT NULL,  
  `vari_id` smallint(6) NOT NULL,  
  `vari` varchar(10) NOT NULL,  
  `isa_id` mediumint(9) NOT NULL,  
  `isa_nimi` varchar(50) NOT NULL,  
  `ema_id` mediumint(9) NOT NULL,  
  `ema_nimi` varchar(50) NOT NULL,  
  `syntpvm` date NOT NULL,  
  PRIMARY KEY (`koira_id`),  
  KEY `id` (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;
```

--

-- Vedos taulusta `koira`

--

```

--
-- Rakenne taululle `lopputulos_parm`
--

CREATE TABLE IF NOT EXISTS `lopputulos_parm` (
  `id` tinyint(2) NOT NULL AUTO_INCREMENT,
  `nimi` varchar(40) NOT NULL,
  `aikaleima` datetime NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=7 ;

--
-- Vedos taulusta `lopputulos_parm`
--

INSERT INTO `lopputulos_parm` (`id`, `nimi`, `aikaleima`) VALUES
(1, 'Hyväksytty', '2012-11-04 00:00:00'),
(2, 'Hyväksytty varauksin', '2012-11-04 23:13:57'),
(3, 'Ei suositella jalostukseen', '2012-11-04 00:00:00'),
(4, 'Hylätty ulkomuodon osalta', '2012-11-04 00:00:00'),
(5, 'Hylätty luonteen osalta', '2012-11-04 00:00:00'),
(6, 'Epävirallinen', '2012-11-04 00:00:00');

-----

--
-- Rakenne taululle `omistaja`
--

CREATE TABLE IF NOT EXISTS `omistaja` (
  `id` mediumint(9) unsigned NOT NULL AUTO_INCREMENT,
  `koira_id` mediumint(9) NOT NULL,

```

```

`nimi` varchar(50) NOT NULL,
`lahiosoite` varchar(50) NOT NULL,
`postinum` varchar(10) NOT NULL,
`postitmp` varchar(30) NOT NULL,
`maa` varchar(50) NOT NULL,
PRIMARY KEY (`id`),
KEY `koira_id` (`koira_id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;

```

```
--
```

```
-- Vedos taulusta `omistaja`
```

```
--
```

```

INSERT INTO `omistaja` (`id`, `koira_id`, `nimi`, `lahiosoite`, `postinum`, `pos-
titmp`, `maa`) VALUES
(1, 1, 'Marja Testinen', 'Osoite 4', '12345', 'Toimipaikka', 'Suomi');

```

```
-----
```

```
--
```

```
-- Rakenne taululle `tarkastus_paikka`
```

```
--
```

```

CREATE TABLE IF NOT EXISTS `tarkastus_paikka` (
  `id` smallint(6) NOT NULL AUTO_INCREMENT,
  `aika` datetime NOT NULL,
  `paikka` varchar(50) NOT NULL,
  `tarkastajat` varchar(100) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;

```

```
-----
```

```

--
-- Rakenne taululle `tulos_parm`
--

CREATE TABLE IF NOT EXISTS `tulos_parm` (
  `id` smallint(3) NOT NULL AUTO_INCREMENT,
  `kohde` varchar(3) NOT NULL,
  `nimi` varchar(50) DEFAULT NULL,
  `tulos` varchar(2) NOT NULL,
  `teksti` varchar(50) DEFAULT NULL,
  PRIMARY KEY (`id`),
  KEY `Kohde` (`kohde`,`tulos`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=232 ;

--
-- Vedos taulusta `tulos_parm`
--

INSERT INTO `tulos_parm` (`id`, `kohde`, `nimi`, `tulos`, `teksti`) VALUES
(26, 'KOK', 'Koko', '1A', 'keskikokoinen'),
(27, 'KOK', 'Koko', '1B', 'alle keskikoon'),
(28, 'KOK', 'Koko', '1C', 'yli keskikoon'),
(29, 'KOK', 'Koko', '2A', 'suurehko'),
(30, 'KOK', 'Koko', '2B', 'turhan pieni'),
(31, 'KOK', 'Koko', '3A', 'liian suuri *'),
(32, 'KOK', 'Koko', '3B', 'mitään tällaista pieni *'),
(33, 'YLE', 'Yleisvaikutelma', '1A', 'erittäin hyvä'),
(34, 'YLE', 'Yleisvaikutelma', '2A', 'hieman neliämiäinen *'),
(35, 'YLE', 'Yleisvaikutelma', '2B', 'hieman raskas *'),
(36, 'YLE', 'Yleisvaikutelma', '2C', 'hieman kevyt *'),
(37, 'YLE', 'Yleisvaikutelma', '2D', 'suhteell. pieni päänä *'),
(38, 'YLE', 'Yleisvaikutelma', '2E', 'maavara hiem. yli 1/3 *'),
(39, 'YLE', 'Yleisvaikutelma', '3A', 'liian raskas !'),

```

- (40, 'YLE', 'Yleisvaikutelma', '3B', 'hintelÄ¤ !'),
- (41, 'YLE', 'Yleisvaikutelma', '3C', 'liian suuri maavara !'),
- (42, 'YLE', 'Yleisvaikutelma', '3D', 'liian matala, kÄ¤mpelÄ¤ !'),
- (43, 'SUK', 'Sukupuolileima', '1A', 'selvÄ¤'),
- (44, 'SUK', 'Sukupuolileima', '2A', 'riittÄ¤vÄ¤'),
- (45, 'SUK', 'Sukupuolileima', '3A', 'heikko *'),
- (46, 'KIV', 'Kivekset', '1A', 'normaalit'),
- (47, 'KIV', 'Kivekset', '3A', 'liian pehmeÄ¤t *'),
- (48, 'KIV', 'Kivekset', '3B', 'liian pienet *'),
- (49, 'KIV', 'Kivekset', '0A', 'toinen puuttuu'),
- (50, 'KIV', 'Kivekset', '0B', 'molemmat puuttuvat'),
- (51, 'PÄ,Ä,,', 'PÄ¤Ä¤', '1A', 'rotum. mukainen'),
- (52, 'PÄ,Ä,,', 'PÄ¤Ä¤', '2A', 'hieman leveÄ¤ kallo-osa'),
- (53, 'PÄ,Ä,,', 'PÄ¤Ä¤', '2B', 'hieman kapea kuono-osa'),
- (54, 'PÄ,Ä,,', 'PÄ¤Ä¤', '2C', 'hieman lyhyt kuono'),
- (55, 'PÄ,Ä,,', 'PÄ¤Ä¤', '2D', 'viettÄ¤vÄ¤ kallo-osa'),
- (56, 'PÄ,Ä,,', 'PÄ¤Ä¤', '2E', 'liikaa otsapengertÄ¤'),
- (57, 'PÄ,Ä,,', 'PÄ¤Ä¤', '2F', 'ei riitt. venyvÄ¤ kita'),
- (58, 'PÄ,Ä,,', 'PÄ¤Ä¤', '2G', 'hieman kevyt alaleuka'),
- (59, 'PÄ,Ä,,', 'PÄ¤Ä¤', '3A', 'karkea ja epÄ¤jalo *'),
- (60, 'PÄ,Ä,,', 'PÄ¤Ä¤', '3B', 'liian kapea ja hento *'),
- (61, 'PÄ,Ä,,', 'PÄ¤Ä¤', '3C', 'kapea alaleuka *'),
- (62, 'KOR', 'Korvat', '1A', 'rotum. mukaiset'),
- (63, 'KOR', 'Korvat', '2A', 'lyhyet'),
- (64, 'KOR', 'Korvat', '2B', 'hieman pienet'),
- (65, 'KOR', 'Korvat', '2C', 'tarpeettoman suuret'),
- (66, 'KOR', 'Korvat', '2D', 'hieman ilmavat'),
- (67, 'KOR', 'Korvat', '2E', 'hieman yläs kiinnittyneet'),
- (68, 'KOR', 'Korvat', '2F', 'hieman alas kiinnittyneet'),
- (69, 'KOR', 'Korvat', '3A', 'liian ilmavat'),
- (70, 'KOR', 'Korvat', '3B', 'liian pienet'),
- (71, 'KOR', 'Korvat', '3C', 'liian kapeat'),
- (72, 'KOR', 'Korvat', '3D', 'laskostuneet'),

- (73, 'SIL', 'SilmÄŕt', '1A', 'rotum. mukaiset'),
 (74, 'SIL', 'SilmÄŕt', '2A', 'hieman suuret'),
 (75, 'SIL', 'SilmÄŕt', '2B', 'hieman pyÄŕreÄŕt'),
 (76, 'SIL', 'SilmÄŕt', '2C', 'liian pienet ja pyÄŕreÄŕt'),
 (77, 'SIL', 'SilmÄŕt', '3A', 'liian suuret ja pyÄŕreÄŕt'),
 (78, 'SIV', 'SilmienvÄŕri', '1A', 'tumma- keskiruskea'),
 (79, 'SIV', 'SilmienvÄŕri', '1B', 'ruskealla keski-vaaleanr.'),
 (80, 'SIV', 'SilmienvÄŕri', '2A', 'vaaleanruskea'),
 (81, 'SIV', 'SilmienvÄŕri', '2B', 'ruskealla kellanruskea'),
 (82, 'SIV', 'SilmienvÄŕri', '2C', 'sin. harm. tai herasilmÄŕt'),
 (83, 'SIV', 'SilmienvÄŕri', '3A', 'kellanruskea'),
 (84, 'SIV', 'SilmienvÄŕri', '3B', 'ruskealla keltaiset'),
 (85, 'SIV', 'SilmienvÄŕri', '0A', 'vÄŕrÄŕrÄŕnvÄŕriset'),
 (86, 'KAV', 'KarvanvÄŕri', '1A', 'rotum. mukainen'),
 (87, 'KAV', 'KarvanvÄŕri', '2A', 'hieman pienet vÄŕrimerkit'),
 (88, 'KAV', 'KarvanvÄŕri', '2B', 'liian suuret vÄŕrimerkit'),
 (89, 'KAV', 'KarvanvÄŕri', '2C', 'liian vaaleat merkit'),
 (90, 'KAV', 'KarvanvÄŕri', '2D', 'hieman nokiset merkit'),
 (91, 'KAV', 'KarvanvÄŕri', '2E', 'hieman valk. rinnassa'),
 (92, 'KAV', 'KarvanvÄŕri', '3A', 'liian nokiset merkit'),
 (93, 'KAV', 'KarvanvÄŕri', '3B', 'suuret valkoiset merkit'),
 (94, 'KAV', 'KarvanvÄŕri', '0A', 'ilman vÄŕrimerkkejÄŕ'),
 (95, 'KAV', 'KarvanvÄŕri', '0B', 'liian paljon valkoista'),
 (96, 'KAV', 'KarvanvÄŕri', '0C', 'tÄŕysin virheell. vÄŕri'),
 (97, 'PIG', 'Pigmentti', '1A', 'musta'),
 (98, 'PIG', 'Pigmentti', '1B', 'ruskealla ruskea'),
 (99, 'PIG', 'Pigmentti', '2A', 'ruskea'),
 (100, 'PIG', 'Pigmentti', '2B', 'ruskealla liian vaalea'),
 (101, 'PIG', 'Pigmentti', '3A', 'lihanvÄŕrinen'),
 (102, 'PUR', 'Purenta', '1A', 'leikkaava'),
 (103, 'PUR', 'Purenta', '1B', 'tasapurenta'),
 (104, 'PUR', 'Purenta', '1C', 'niukka'),
 (105, 'PUR', 'Purenta', '2A', 'hieman kapea'),

- (106, 'PUR', 'Purenta', '2B', 'hieman epätasainen'),
- (107, 'PUR', 'Purenta', '3A', 'hieman ahdas (kulmah.) *'),
- (108, 'PUR', 'Purenta', '0A', 'ahdas (kulmah. kital.)'),
- (109, 'PUR', 'Purenta', '0B', 'yläpurenta'),
- (110, 'PUR', 'Purenta', '0C', 'alapurenta'),
- (111, 'PUR', 'Purenta', '0D', 'vino purenta'),
- (112, 'HAM', 'Hampaat', '1A', 'kaikki hampaat'),
- (113, 'HAM', 'Hampaat', '1B', '1-2 P1 puuttuu'),
- (114, 'HAM', 'Hampaat', '1C', 'M3 puuttuu'),
- (115, 'HAM', 'Hampaat', '2A', 'useita P1 ja M3 puuttuu'),
- (116, 'HAM', 'Hampaat', '3A', 'muu kuin P1/M3 puutt. *'),
- (117, 'HAM', 'Hampaat', '0A', 'useita hampaita puuttuu'),
- (118, 'KAU', 'Kaula', '1A', 'rotum. mukainen'),
- (119, 'KAU', 'Kaula', '2A', 'hieman lyhyt'),
- (120, 'KAU', 'Kaula', '2B', 'hieman paksu'),
- (121, 'KAU', 'Kaula', '2C', 'hieman kaulanahkaa'),
- (122, 'KAU', 'Kaula', '2D', 'hieman ohut'),
- (123, 'KAU', 'Kaula', '3A', 'lyhyt ja paksu *'),
- (124, 'KAU', 'Kaula', '3B', 'paljon kaulanahkaa *'),
- (125, 'SEL', 'Selkälinja', '1A', 'korkea ja pitkä selkä'),
- (126, 'SEL', 'Selkälinja', '2A', 'matala selkä'),
- (127, 'SEL', 'Selkälinja', '2B', 'liikaa kaart. lanne-osa'),
- (128, 'SEL', 'Selkälinja', '2C', 'liian suora selkä'),
- (129, 'SEL', 'Selkälinja', '2D', 'hieman painunut selkä'),
- (130, 'SEL', 'Selkälinja', '2E', 'hieman takakorkea'),
- (131, 'SEL', 'Selkälinja', '3A', 'voim. käälyrist. selkä *'),
- (132, 'SEL', 'Selkälinja', '3B', 'notkoselkä *'),
- (133, 'SEL', 'Selkälinja', '3C', 'selvästi takakorkea *'),
- (134, 'VAT', 'Vatsalinja', '1A', 'rotum. mukainen'),
- (135, 'VAT', 'Vatsalinja', '1B', 'nuorella hiem. yläs vet.'),
- (136, 'VAT', 'Vatsalinja', '2A', 'liian yläs vetäytynyt'),
- (137, 'RIN', 'Rintakehä', '1A', 'pitkä, tilava..+ eturinta'),
- (138, 'RIN', 'Rintakehä', '2A', 'liian pyöreä, tynnyrim.'),

- (139, 'RIN', 'RintakehÃ¤', '2B', 'liian kapea, litteÃ¤'),
- (140, 'RIN', 'RintakehÃ¤', '2C', 'pieni eturinta'),
- (141, 'RIN', 'RintakehÃ¤', '2D', 'ei riittÃ¤vÃ¤n syvÃ¤'),
- (142, 'RIN', 'RintakehÃ¤', '2E', 'liian syvÃ¤'),
- (143, 'RIN', 'RintakehÃ¤', '2F', 'liian lyhyt'),
- (144, 'RIN', 'RintakehÃ¤', '0A', 'epÃ¤muodostunut'),
- (145, 'ETR', 'Eturaajat', '1A', 'rotum. mukaiset'),
- (146, 'ETR', 'Eturaajat', '2A', 'niukat kulmaukset'),
- (147, 'ETR', 'Eturaajat', '2B', 'lyhyt olkavarsi'),
- (148, 'ETR', 'Eturaajat', '2C', 'etuasent. pysty lapa'),
- (149, 'ETR', 'Eturaajat', '2D', 'hieman kÃ¤yrÃ¤t'),
- (150, 'ETR', 'Eturaajat', '2E', 'liev. ulkokiert. kÃ¤pÃ¤lÃ¤t'),
- (151, 'ETR', 'Eturaajat', '2F', 'leveÃ¤asentoiset'),
- (152, 'ETR', 'Eturaajat', '2G', 'ahdasasentoiset'),
- (153, 'ETR', 'Eturaajat', '2H', 'heiman lÃ¤ÿysÃ¤t kyynÃ¤rpÃ¤Ã¤t'),
- (154, 'ETR', 'Eturaajat', '2I', 'hieman taipuvat ranteet'),
- (155, 'ETR', 'Eturaajat', '3A', 'liian kÃ¤yrÃ¤t *'),
- (156, 'ETR', 'Eturaajat', '3B', 'ulkokierteiset kÃ¤pÃ¤lÃ¤t *'),
- (157, 'ETR', 'Eturaajat', '3C', 'lÃ¤ÿysÃ¤t kyynÃ¤rpÃ¤Ã¤t *'),
- (158, 'ETR', 'Eturaajat', '3D', 'lÃ¤ÿysÃ¤t lavat *'),
- (159, 'ETR', 'Eturaajat', '0A', 'eteenp. taittavat ranteet'),
- (160, 'LAN', 'Lantio', '1A', 'pitkÃ¤, leveÃ¤, pyÃ¤reÃ¤..'),
- (161, 'LAN', 'Lantio', '2A', 'melko suora lantio'),
- (162, 'LAN', 'Lantio', '2B', 'melko lyhyt lantio'),
- (163, 'LAN', 'Lantio', '2C', 'melko luisu lantio'),
- (164, 'LAN', 'Lantio', '2D', 'hieman pitkÃ¤ lanne'),
- (165, 'LAN', 'Lantio', '3A', 'liian luisu lantio *'),
- (166, 'TAR', 'Takaraajat', '1A', 'rotum. mukaiset'),
- (167, 'TAR', 'Takaraajat', '2A', 'niukat kulmaukset'),
- (168, 'TAR', 'Takaraajat', '2B', 'lievÃ¤sti pihtiasentoiset'),
- (169, 'TAR', 'Takaraajat', '2C', 'lievÃ¤sti lÃ¤nkisÃ¤Ã¤riset'),
- (170, 'TAR', 'Takaraajat', '2D', 'kapeat reidet'),
- (171, 'TAR', 'Takaraajat', '2E', 'hieman korkea kinner'),

- (172, 'TAR', 'Takaraajat', '2F', 'hieman lyhyt sÃ¤Ã¤ri'),
 (173, 'TAR', 'Takaraajat', '3A', 'huonosti kulmautuneet *'),
 (174, 'TAR', 'Takaraajat', '3B', 'ahtaata ja pihtiasent. *'),
 (175, 'TAR', 'Takaraajat', '3C', 'selvÃ¤sti lÃ¤nkisÃ¤Ã¤riset *'),
 (176, 'TAR', 'Takaraajat', '3D', 'heikot ja epÃ¤varmat *'),
 (177, 'HÃ¤,,N', 'HÃ¤ntÃ¤', '1A', 'rotum. mukainen'),
 (178, 'HÃ¤,,N', 'HÃ¤ntÃ¤', '2A', 'hieman korkea asento'),
 (179, 'HÃ¤,,N', 'HÃ¤ntÃ¤', '2B', 'liian ohut'),
 (180, 'HÃ¤,,N', 'HÃ¤ntÃ¤', '2C', 'liian pitkÃ¤'),
 (181, 'HÃ¤,,N', 'HÃ¤ntÃ¤', '2D', 'suhteettoman lyhyt'),
 (182, 'HÃ¤,,N', 'HÃ¤ntÃ¤', '2E', 'liian yllÃ¤s kiinnittynyt'),
 (183, 'HÃ¤,,N', 'HÃ¤ntÃ¤', '2F', 'liian alas kiinnittynyt'),
 (184, 'HÃ¤,,N', 'HÃ¤ntÃ¤', '3A', 'pysty seist. & liikk. *'),
 (185, 'HÃ¤,,N', 'HÃ¤ntÃ¤', '3B', 'selÃ¤n yli kaartuva *'),
 (186, 'HÃ¤,,N', 'HÃ¤ntÃ¤', '0A', 'nikamavika'),
 (187, 'KÃ¤,,P', 'KÃ¤pÃ¤lÃ¤t', '1A', 'pyÃ¶reÃ¤t, tiiviit'),
 (188, 'KÃ¤,,P', 'KÃ¤pÃ¤lÃ¤t', '2A', 'hieman pitkÃ¤t'),
 (189, 'KÃ¤,,P', 'KÃ¤pÃ¤lÃ¤t', '2B', 'hieman litteÃ¤t'),
 (190, 'KÃ¤,,P', 'KÃ¤pÃ¤lÃ¤t', '2C', '5 varvas poist. etukÃ¤p. '),
 (191, 'KÃ¤,,P', 'KÃ¤pÃ¤lÃ¤t', '3A', 'lÃ¤piastuja *'),
 (192, 'KÃ¤,,P', 'KÃ¤pÃ¤lÃ¤t', '3B', 'hajavarpaat *'),
 (193, 'LUU', 'Luusto', '1A', 'vahva'),
 (194, 'LUU', 'Luusto', '1B', 'riittÃ¤vÃ¤'),
 (195, 'LUU', 'Luusto', '2A', 'hieman hento'),
 (196, 'LUU', 'Luusto', '3A', 'liian hento *'),
 (197, 'KAR', 'Karvapeite', '1A', 'rotum. mukainen'),
 (198, 'KAR', 'Karvapeite', '2A', 'vain vÃ¤hÃ¤n huomautt. -'),
 (199, 'KAR', 'Karvapeite', '3A', 'paljon huomauttamista *'),
 (200, 'LII', 'Liikunta', '1A', 'rotum. mukainen'),
 (201, 'LII', 'Liikunta', '2A', 'hieman lyhyt askel'),
 (202, 'LII', 'Liikunta', '2B', 'hieman leveÃ¤t etuliikk. '),
 (203, 'LII', 'Liikunta', '2C', 'hieman ahtaata etuliikk. '),
 (204, 'LII', 'Liikunta', '2D', 'hieman ahtaata takaliikk. '),

(205, 'LII', 'Liikunta', '2E', 'hieman leveÃ¤t takaliikk.'),
 (206, 'LII', 'Liikunta', '2F', 'kinnerahtaat takaliikk.'),
 (207, 'LII', 'Liikunta', '3A', 'tehottomat liikkeet *'),
 (208, 'LII', 'Liikunta', '3B', 'epÃ¤suhtaiset liikkeet *'),
 (209, 'KUN', 'Kunto', '1A', 'normaali'),
 (210, 'KUN', 'Kunto', '2A', 'laiha'),
 (211, 'KUN', 'Kunto', '2B', 'lihava'),
 (212, 'KUN', 'Kunto', '3A', 'liian laiha *'),
 (213, 'KUN', 'Kunto', '3B', 'liian lihava *'),
 (214, 'LUO', 'Luonne', '1A', 'rauhallinen'),
 (215, 'LUO', 'Luonne', '1B', 'vilkas ja iloinen'),
 (216, 'LUO', 'Luonne', '1C', 'avoin ja ystÃ¤vÃ¤llinen'),
 (217, 'LUO', 'Luonne', '2A', 'hieman pidÃ¤ttyvÃ¤inen *'),
 (218, 'LUO', 'Luonne', '3A', 'liian pidÃ¤ttyvÃ¤inen !'),
 (219, 'LUO', 'Luonne', '3B', 'hieman arka !'),
 (220, 'LUO', 'Luonne', '0A', 'arka'),
 (221, 'LUO', 'Luonne', '0B', 'vihainen'),
 (222, 'PÃ¤,,Ã¤,,', 'PÃ¤Ã¤', '2H', 'Hieman voimakkaat poskiluut'),
 (223, 'PIG', 'Pigmentti', '2C', 'Ruskeata kynsissÃ¤'),
 (224, 'RIN', 'RintakehÃ¤', '2G', 'Liian lyhyt rintalasta'),
 (225, 'RIN', 'RintakehÃ¤', '2H', 'rintakehÃ¤n syvin kohta eturaajojen takana'),
 (230, 'LIK', 'Lihaskunto', '1A', 'hyvÃ¤'),
 (231, 'LIK', 'Lihaskunto', '1B', 'riittÃ¤vÃ¤');

--

-- Rajoitteet vedostauluille

--

--

-- Rajoitteet taululle `omistaja`

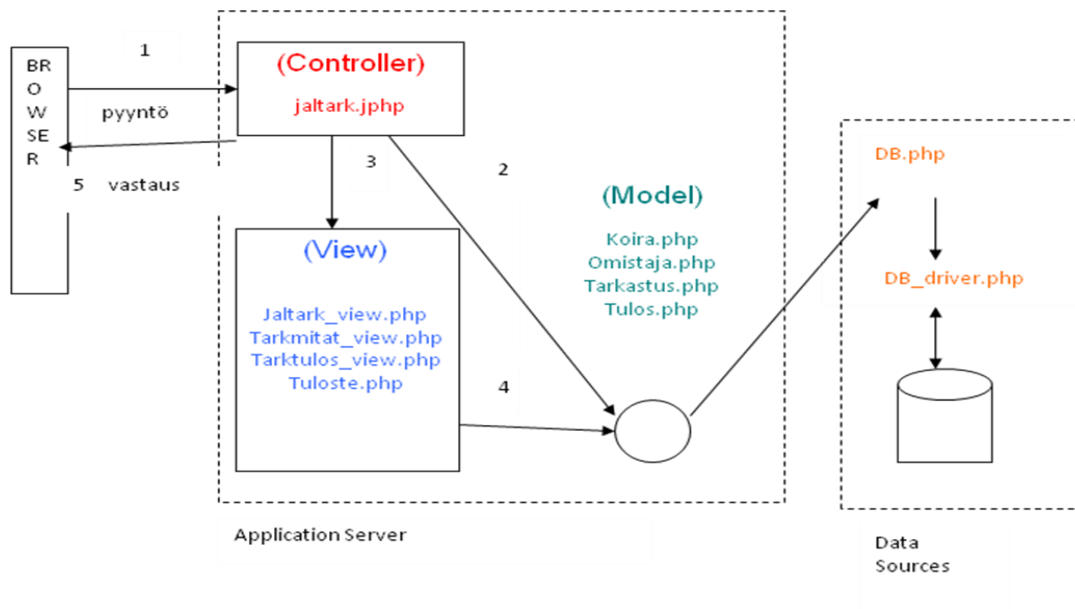
--

ALTER TABLE `omistaja`

ADD CONSTRAINT `omistaja_ibfk_1` FOREIGN KEY (`koira_id`) REFERENCES `koira` (`koira_id`);

5. Arkkitehtuurimalli

Jalostustarkastus sovelluksen MVC malli. Käytetään kolmitasoista mallia: ohjausluokat, liiketoimintaluokat, tietokantaluokat.



6. Yhteistyökaaviot

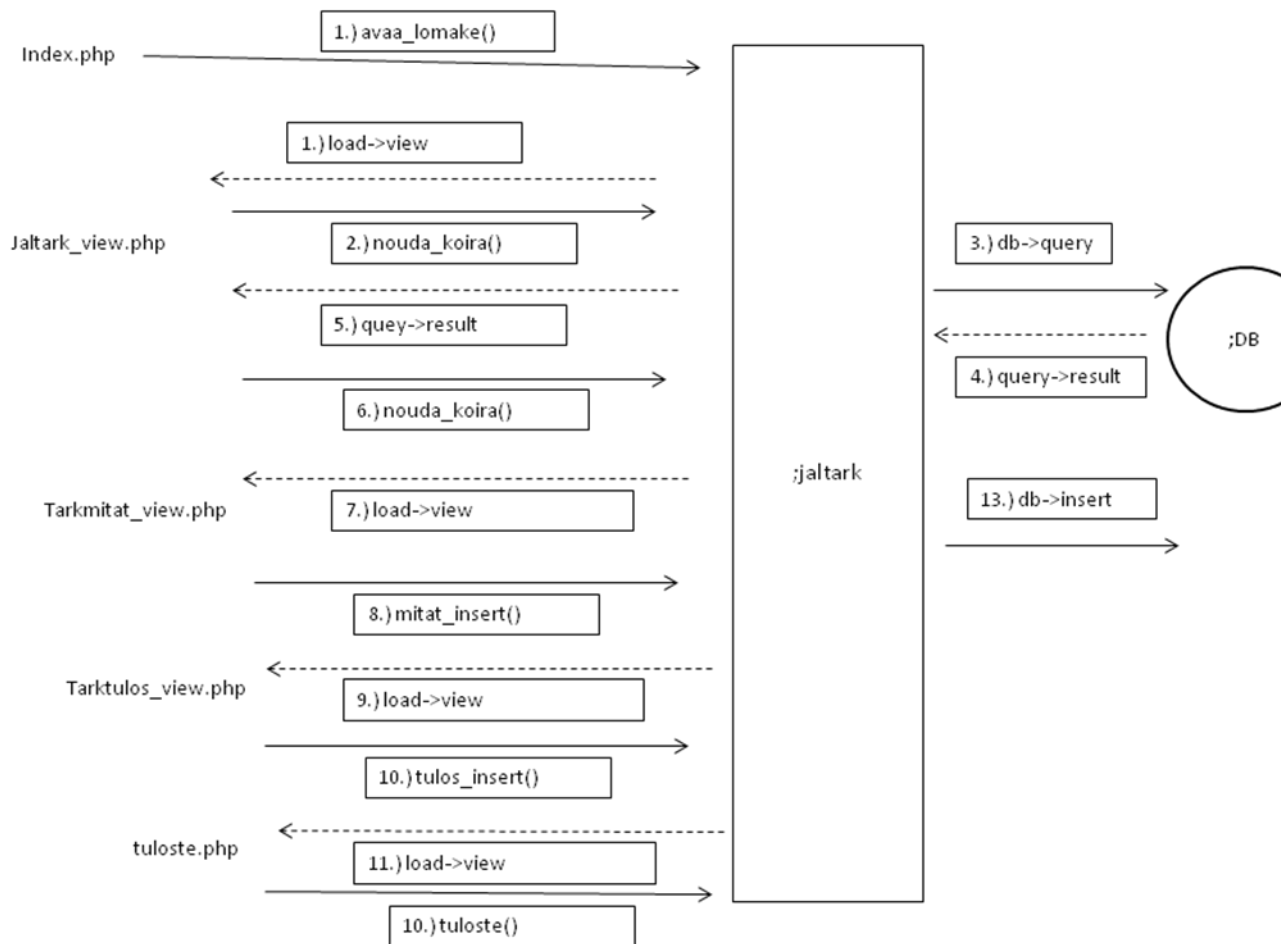
koiraHaku

- 1 Etusivulta (`index.php`) linkistä "Hae koira tietokannasta" ohjaututaan `noudakoirat_view.php` -sivulle.
- 2 `noudakoirat_view.php` lähettää tiedot maykky-oliolle.
- 3 Maykky-olio tarkastaa asiakkaan syöttämät tiedot ja lähettää ne Database-oliolle (Sql query)
- 4 Database-olio on hyväksynyt tiedot. Tiedot on syötetty oikein. Database-palauttaa kyselytuloksena listan Maykky-oliolle (query → result).
- 5 Maykky-olio välittää listan `noudakoirat_view.php` -sivulle (suoritusOk).
- 6 Lataa Koirat-nappia painamalla `koiraHaku.php` -sivu lähettää listan maykky-oliolle.
- 7 Maykky-olio lähettää listan Database-oliolle (Sql query).

- 8 Database-olio on hyväksynyt tiedot. Tiedot kopioituvat Jalostustarkastus tietokantaan.

Jalostustarkastus

- 1 Etusivulta (index.php) linkistä ”Uusi Jalostustarkastus” jaltark-olio ohjaa näkymään jaltark_view.
- 2 jaltark_view.php lähettää tiedot Jaltark-oliolle(nouda_koira).
- 3 Jaltark-olio lähettää kyselyn database-oliolle suoritettavaksi(query).
- 4 Database palauttaa kyselytuloksena listan Jaltark-oliolle(nouda_koira).
- 5 Jaltark-olio välittää kyselyntuloksen näkymälle jaltark_view.
- 6 Jaltark_view lähettää tiedot Jaltark-oliolle (nouda_koira)
- 7 Jaltark-olio suorittaa kenttientietojen tarkastuksen ja ohjaa sivulle tarkmitat_view.php.
- 8 Tarkmitat_view.php lähettää tiedot jaltark-oliolle(mitat_insert)
- 9 Jaltark-olio tarkastaa kenttien tiedot ja ohjaa sivulle tarktulos_view.php.
- 10 Tarktulos_view.php lähettää tiedot jaltark-oliolle(tulos_insert).
- 11 Jaltark-olio tarkastaa kenttien tiedot ja ohjaa sivulle tuloste.php.
- 12 Tuloste.php lähettää pyynnön jaltark-oliolle(tuloste)
- 13 Jaltark-olio lähettää kyselyn database-oliolle(insert)
- 14 Jaltark-olio suorittaa pyynnön(tulosta)

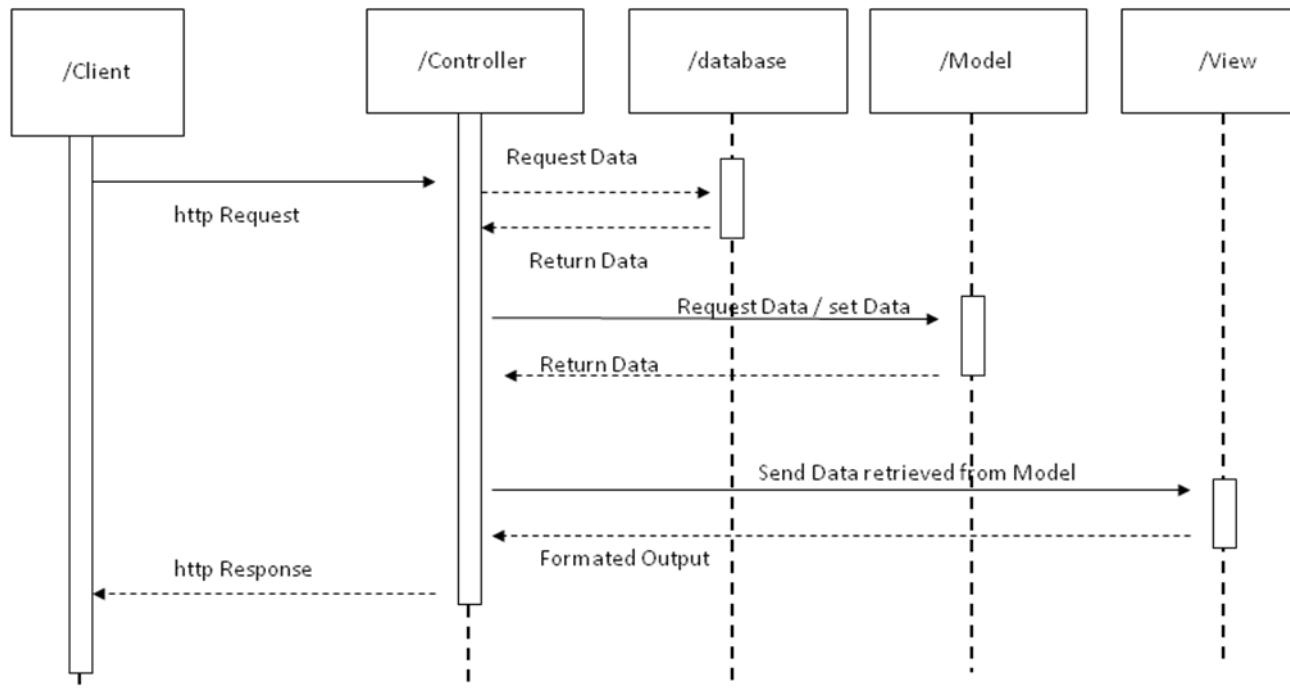


Kuva 11. Yhteistyökaavio Jalostustarkastus

SiirraKantaan

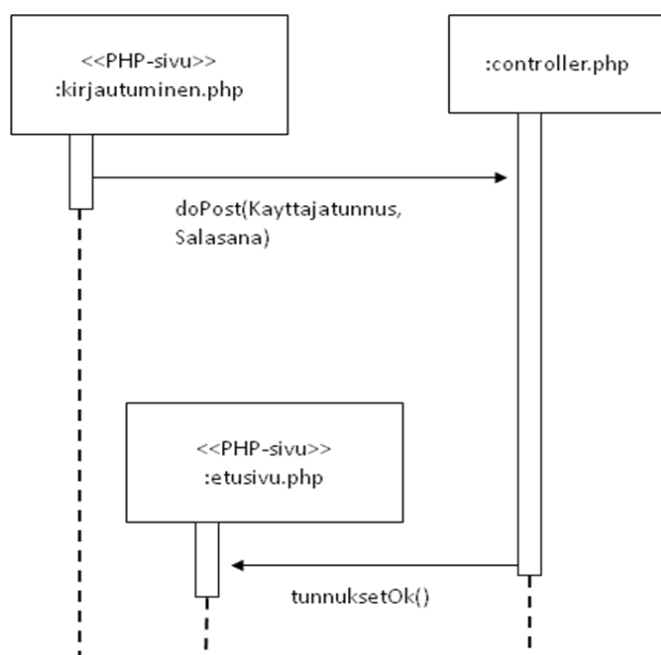
- 1 Etusivulta (index.php) linkistä ”Siirrä tarkastukset” ohjaututaan siirraKoira_view.php -sivulle.
- 2 SiirraKoira_view.php lähettää kyselyn maykky-oliolle(siirraKanta).
- 3 Maykky-olio lähettää kyselyn database-oliolle tarkastettavaksi (query)
- 4 database-palauttaa kyselytuloksena listan Controllerille(suoritusOk).
- 5 Maykky-olio lähettää listan Database-oliolle (suoritaSqlLause).
- 6 Päivitys-olio on hyväksynyt tiedot. Tiedot kopioituvat maykkynet tietokantaan.

7. Viestiyhteyskaaviot

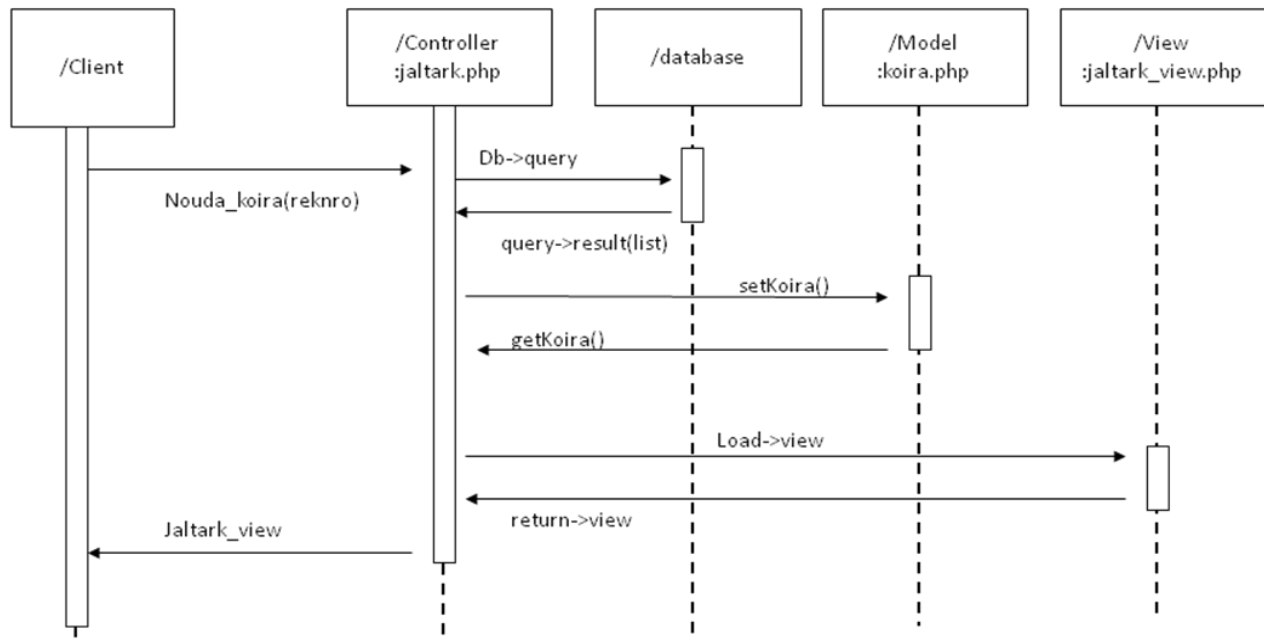


Kuva 12. Yleiskuvas MVC mallin viestiyhteyskaaviosta PHP:ssä

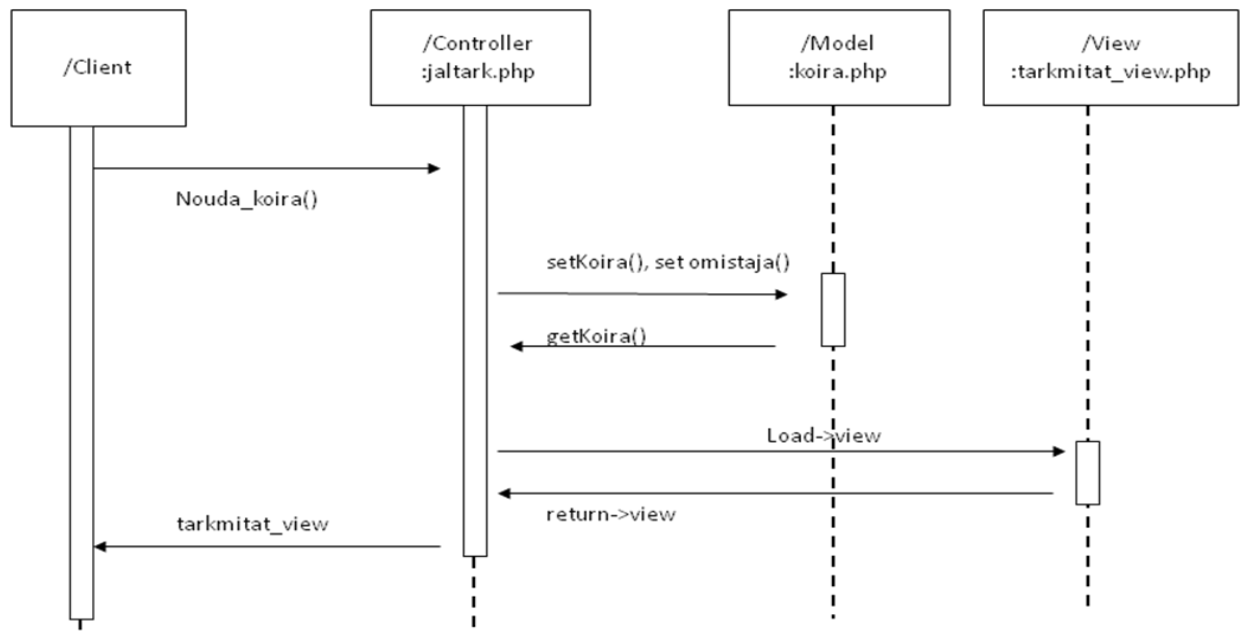
Kuvan 12 mukaan toimii tämänkin sovelluksen viestinvälitys eri osien välillä. Alla on vielä kuvattuna jokaisesta yhteistyökaaviosta viestiyhteyskaaviot.



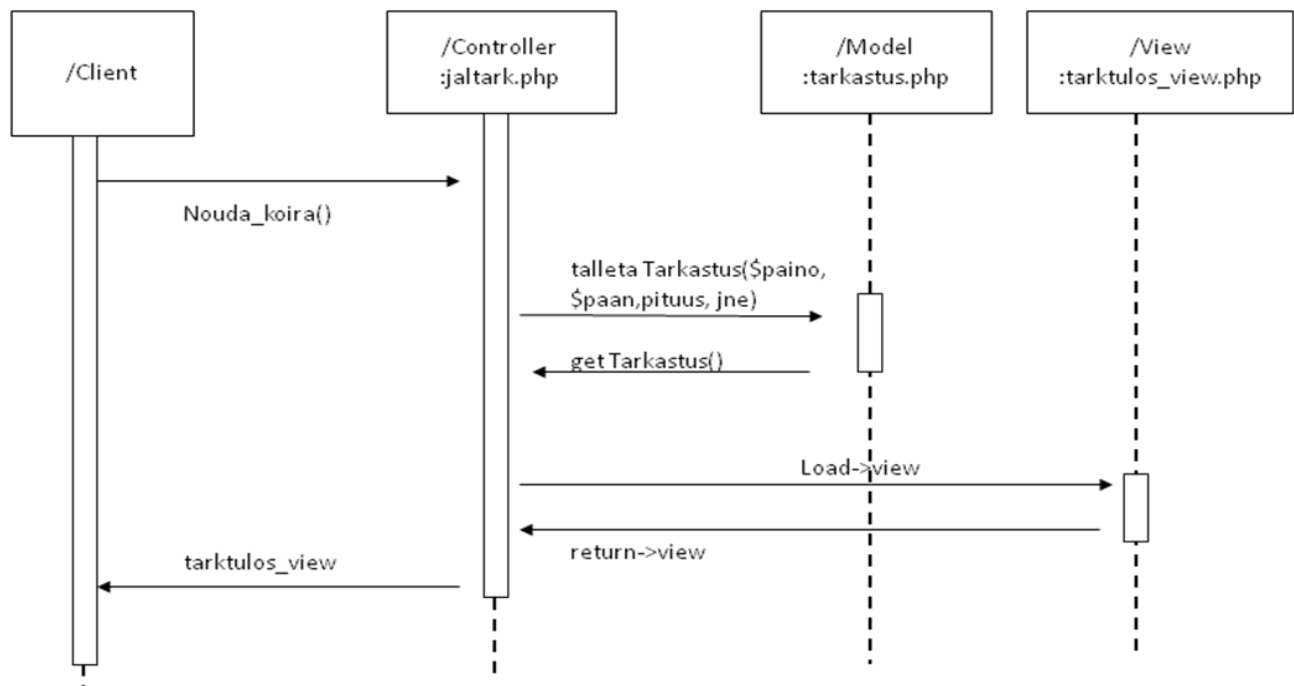
Kuva 13. Kirjautumisen viestiyhteyskaavio



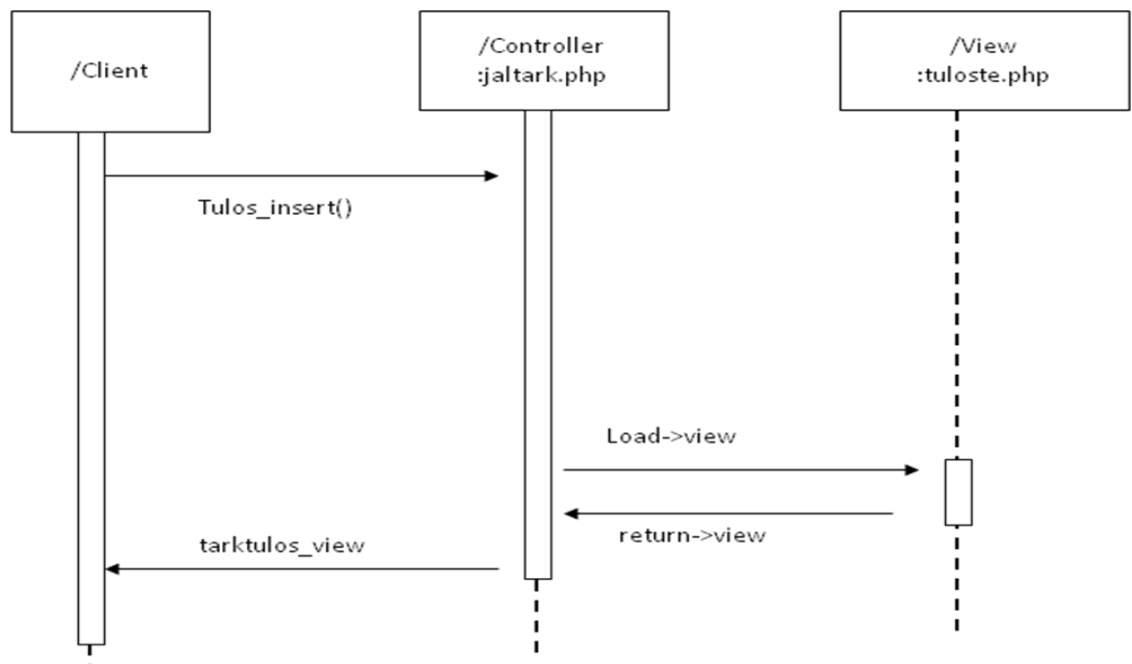
Kuva 14. Jalostustarkastus lomakkeelle noudetaan koiran tiedot



Kuva 15. Siirrytään näkymään tarkmitat_view

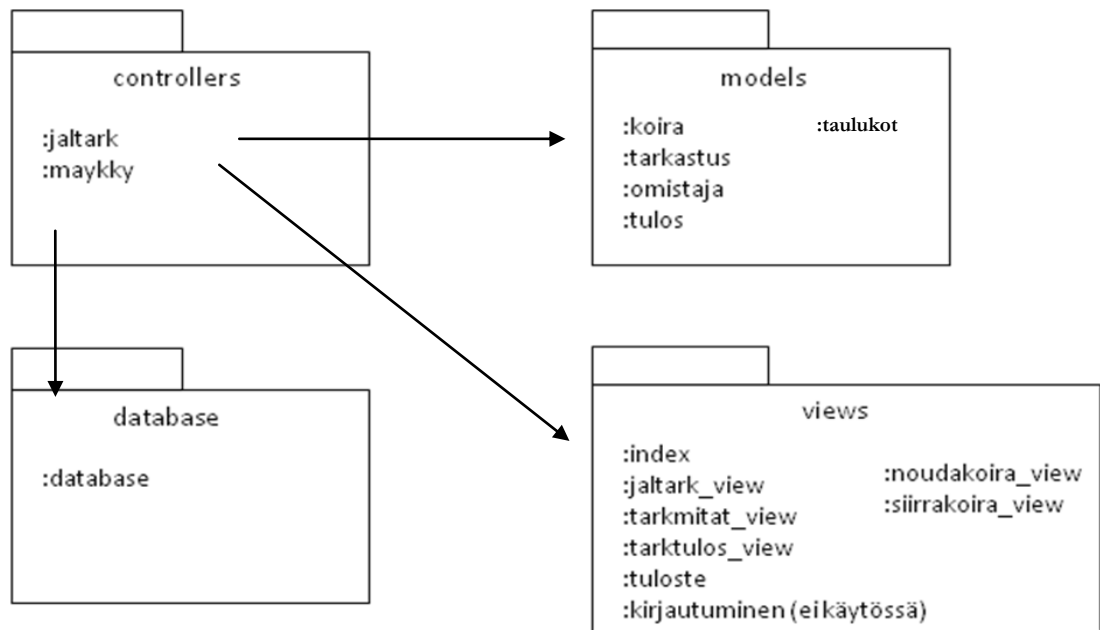


Kuva 16. Siirrytään näkymään tarktulos_view.



Kuva 17. Siirrytään tulostusnäkymään

8. Paketit



Kuva 18. Sovelluksen pakettirakenne

Liite 3. Aikataulu toteutma

teht.no.	tehtävä	lopputulos	aloituskriteeri	tuntia	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49
1	Projektin käynnistys	hyväksytty projektisuunnitelma	hankeresurssi on myönnetty	25,5																											
1.1	Projektisuunnitelma	Valmis projektisuunnitelma ehdotus	hankeresurssi on myönnetty	10																											
1.2	Aloituskokouksen valmistelu	esityslista ja materiaali jaettu	projektisuunnitelma	2																											
1.3	Aloituskokous	käynnistetty projekti	projektisuunnitelma	1,5																											
1.4	Pöytäkirjan laatiminen	pöytäkirja jaettu	kokous on pidetty	2																											
1.5	Määrittysten läpikäynti		määrittelyt toimeksiantajalta	10																											
2	Projektin ohjaus	hallittu edistyminen	projekti on käynnistetty	28																											
2.1	Edistymisraportti	raportti		20																											
2.2	Ohjauskokouksen valmistelu			4																											
2.3	Ohjauskokous	esityslista, materiaali jaettu	sovittu kokousaika (3 kpl)	2																											
2.4	Pöytäkirjan laatiminen	pöytäkirjat (3 kpl)	kokous on pidetty	2																											
3	Projektin teko	hallittu edistyminen	projekti on käynnistetty	209																											
3.1	Työn suunnittelu	muistio	sovittu palaveri	45																											
3.2	Työn toteutus	muistio	sovittu palaveri	152																											
3.3	Korjaukset + viimeistely	valmis työ	ohjauskokous pidetty	10																											
3.4	Työn testaus	testausraportti	työn toteutus valmis	2																											
4	Dokumentaatio	hallittu edistyminen	projekti on käynnistetty	42																											
4.1	Opinnäytetyö	dokumentaatio		32																											

[illegible]

Liite 4. Testauslomakkeet

1

Käyttötapaus: Onnistunut jalostustarkastus

Testaaja:

Pvm:

No	Testitapauksen kuvaus	Odotettu tulos	Poikkeamat	Ok/hylätty
1	Testaaja käynnistää ohjelman	Avautuu Jalostustarkastuksen etusivu, jossa kolme linkkiä: <ul style="list-style-type: none">- Tee tarkastus- Siirrä koirat- hae koira		
2a	Testaaja painaa etusivun linkkiä Tee tarkastus	Avautuu lomake, jossa on syöttökentät koiran ja omistajan tiedoille: reknro rotu nimi sukupuoli väri isän reknro isän nimi		

		emän reknro emän nimi syntymäpäivä(syntpvm) omistaja lahiosoite postinum postitmp maa <u>Aktiiviset nappulat:</u> Hae koira, Jatka, Peruuta		
2b	Tutkitaan, saako koiran hattua rekisterinumerolla (reknro testiä varten 1234565-8): syötetään reknro kenttään koiran rekisterinnumero ja painetaan Hae koira-painiketta	Koira löytyy kannasta ja tiedot tulevat lomakkeen kenttiin.		
3	Testaaja siirtyy eteenpäin klikkaamalla Jatka-painiketta	Avautuu uusi lomake, jossa kentät koiran mittojen syöttämistä varten:		

		paino pään pituus rinnanympärys säkäkorkeus maavara rungon syvyys (vain lukuoikeus) rungon pituus näyttelykäynnit (valintalista) <u>Aktiiviset nappulat:</u> Edellinen, Jatka, Peruuta		
4a	Testaaja syöttää arvot seuraaviin kenttiin ja valitsee valintalistasta haluamansa arvon: paino pään pituus rinnanympärys säkäkorkeus maavara rungon pituus näyttelykäynnit (valintalista)	Kaikissa kentissä on jokin arvo ja kentän rungon syvyys arvo on muodostunut automaattisesti: $\text{rungon syvyys} = \text{säkäkorkeus} - \text{maavara}.$		

4b	Testaaja siirtyy eteenpäin klikkaamalla Jatka-painiketta	<p>Avautuu uusi lomake, jossa checkbox listaus tarkistustuloksien valintaa varten:</p> <p>koko;</p> <p>yleisvaikutelma;</p> <p>sukupuolileima;</p> <p>kivekset;</p> <p>paa;</p> <p>purenta;</p> <p>hampaat;</p> <p>korvat;</p> <p>silmat;</p> <p>silmienvari;</p> <p>karvanvari;</p> <p>pigmentti;</p> <p>kaula;</p> <p>eturaajat;</p> <p>rintakeha;</p> <p>lantio;</p>		

		takaraajat; hanta; kapalat; luusto; karvapeite; selkalinja; vatsalinja; liikunta; kunto; lihaskunto; luonne; Tekstikenttä: kuvaus Sekä valinnat: aika paikka tarkastajat lopputulos		
--	--	--	--	--

		<u>Aktiiviset nappulat:</u> Edellinen, Jatka, Peruuta		
4c	Testaaja merkitsee jokaiseen kohtaan lomakkeella jonkin arvon	Merkinnät lomakkeelle onnistuvat		
4d	Testaaja siirtyy eteenpäin klikkaamalla Jatka-painiketta	Avautuu tuloste sivu, jossa listattuna kaikki kohteet ja syötetyt arvot <u>Aktiiviset nappulat:</u> Tallenna ja tulosta, Edellinen, Peruuta		
5	Käyttäjä tallettaa ja tulostaa tarkastusraportin klikkaamalla Tallenna ja tulosta -painiketta	Lomake avautuu pdf-tiedostona käyttäjälle. Ilmoitus onnistuneesta kantaan lisäyksestä.		

Käyttötapaus: Jalostustarkastus virhetilanteet

Testaaja:

Pvm:

No	Testitapauksen kuvaus	Odotettu tulos	Poikkeamat	Ok/hylätty
1	Testaaja käynnistää ohjelman	Avautuu Jalostustarkastuksen etusivu, jossa kolme linkkiä: <ul style="list-style-type: none">- Tee tarkastus- Siirrä koirat- hae koira		
2a	Testaaja painaa etusivun linkkiä Tee tarkastus	Avautuu lomake, jossa on syöttökentät koiran ja omistajan tiedoille: reknro rotu nimi sukupuoli väri isän reknro isän nimi emän reknro emän nimi syntymäpäivä(syntpvm)		

		omistaja lahiosoite postinum postitmp maa <u>Aktiiviset nappulat:</u> Hae koira, Jatka, Peruuta		
2b_1	Tutkitaan voiko hakea virheellisellä/ei olemassa olevalla rekisterinumerolla	Sovellus ei löydä koira tietokannasta ja antaa virheilmoituksen.		
2b_2	Tutkitaan, saako koiran hattua rekisterinumerolla (reknro testiä varten 1234565-8): syötetään reknro kenttään koiran rekisterinnumero ja painetaan Hae koira-painiketta	Koira löytyy kannasta ja tiedot tulevat lomakkeen kenttiin.		
2c_1	Tarkistetaan kenttien tarkistuk-	Sovelluksessa ei pääse eteenpäin,		

	set:reknro: jätetään kenttä tyhjäksi ja painetaan Jatka-painiketta	koiran tiedot lomake sivulla auki ja kyseisen kentän vieressä on virheilmoitus		
2c_2	Tarkistetaan kenttien tarkistukset:nimi jätetään kenttä tyhjäksi ja painetaan Jatka-painiketta	Sovelluksessa ei pääse eteenpäin, koiran tiedot lomake sivulla auki ja kyseisen kentän vieressä on virheilmoitus		
2c_3	Tarkistetaan kenttien tarkistukset:sukupuoli, jätetään kenttä tyhjäksi ja painetaan Jatka-painiketta	Sovelluksessa ei pääse eteenpäin, koiran tiedot lomake sivulla auki ja kyseisen kentän vieressä on virheilmoitus		
2c_4	Tarkistetaan kenttien tarkistukset:isa_reknro, jätetään kenttä tyhjäksi ja painetaan Jatka-painiketta	Sovelluksessa ei pääse eteenpäin, koiran tiedot lomake sivulla auki ja kyseisen kentän vieressä on virheilmoitus		
2c_5	Tarkistetaan kenttien tarkistukset:isa_nimi, jätetään kenttä tyhjäksi ja painetaan Jatka-painiketta	Sovelluksessa ei pääse eteenpäin, koiran tiedot lomake sivulla auki ja kyseisen kentän vieressä on virheilmoitus		
2c_6	Tarkistetaan kenttien tarkistukset:	Sovelluksessa ei pääse eteenpäin,		

	set:ema_reknro, jätetään kenttä tyhjäksi ja painetaan Jatka-painiketta	koiran tiedot lomake sivulla auki ja kyseisen kentän vieressä on virheilmoitus		
2c_7	Tarkistetaan kenttien tarkistukset:ema_nimi jätetään kenttä tyhjäksi ja painetaan Jatka-painiketta	Sovelluksessa ei pääse eteenpäin, koiran tiedot lomake sivulla auki ja kyseisen kentän vieressä on virheilmoitus		
2c_8	Tarkistetaan kenttien tarkistukset:syntpvm jätetään kenttä tyhjäksi ja painetaan Jatka-painiketta	Sovelluksessa ei pääse eteenpäin, koiran tiedot lomake sivulla auki ja kyseisen kentän vieressä on virheilmoitus		
2c_9	Testaaja hakee koiran rekisterinumerolla (reknro testiä varten 1234565-8): syötetään reknro kenttään koiran rekisterinumero ja painetaan Hae koira-painiketta	Koira löytyy kannasta ja tiedot tulevat lomakkeen kenttiin.		
2d	Testaaja siirtyy eteenpäin klikkaamalla Jatka-painiketta	Avautuu uusi lomake, jossa kentät koiran mittojen syöttämistä varten: paino pään pituus		

		rinnanympärys säkäkorkeus maavara rungon syvyys (vain lukuoikeus) rungon pituus näyttelykäynnit (valintalista) <u>Aktiiviset nappulat:</u> Edellinen, Jatka, Peruuta		
3a_1	Tarkistetaan kenttien tarkistukset:paino, jätetään kenttä tyhjäksi ja painetaan Jatka-painiketta	Sovelluksessa ei pääse eteenpäin, mittojen syöttö lomake sivulla auki ja kyseisen kentän vieressä on virheilmoitus		
3a_2	Tarkistetaan kenttien tarkistukset:pään pituus, jätetään kenttä tyhjäksi ja painetaan Jatka-painiketta	Sovelluksessa ei pääse eteenpäin, mittojen syöttö lomake sivulla auki ja kyseisen kentän vieressä on virheilmoitus		
3a_3	Tarkistetaan kenttien tarkistukset:rinnanympärys, jätetään kenttä tyhjäksi ja paine-	Sovelluksessa ei pääse eteenpäin, mittojen syöttö lomake sivulla auki ja kyseisen kentän vieressä on virheil-		

	taan Jatka-painiketta	moitus		
3a_4	Tarkistetaan kenttien tarkistukset:säkäkorkeus, jätetään kenttä tyhjäksi ja painetaan Jatka-painiketta	Sovelluksessa ei pääse eteenpäin, mittojen syöttö lomake sivulla auki ja kyseisen kentän vieressä on virheilmoitus		
3a_5	Tarkistetaan kenttien tarkistukset:maavara, jätetään kenttä tyhjäksi ja painetaan Jatka-painiketta	Sovelluksessa ei pääse eteenpäin, mittojen syöttö lomake sivulla auki ja kyseisen kentän vieressä on virheilmoitus		
3a_6	Tarkistetaan kenttien tarkistukset:rungon pituus, jätetään kenttä tyhjäksi ja painetaan Jatka-painiketta	Sovelluksessa ei pääse eteenpäin, mittojen syöttö lomake sivulla auki ja kyseisen kentän vieressä on virheilmoitus		
3a_7	Tarkistetaan kenttien tarkistukset:näyttelykäynnit, jätetään arvoksi ”– Valinta –” ja painetaan Jatka-painiketta	Sovelluksessa ei pääse eteenpäin, mittojen syöttö lomake sivulla auki ja kyseisen kentän vieressä on virheilmoitus		
3b_1	Testaaja syöttää lomakkeen kenttiin kunnolliset arvot, huom: mitat pisteellä(.) eroteltuna.	Syöttö lomakkeen muihin kenttiin onnistuu, paitsi rungon syvyys, johon on vain lukuoikeudet.		

3b_2	Testaaja siirtyy eteenpäin klikkaamalla Jatka-painiketta	<p>Avautuu uusi lomake, jossa check-box listaus tarkistustuloksien valintaa varten:</p> <p>koko;</p> <p>yleisvaikutelma;</p> <p>sukupuolileima;</p> <p>kivekset;</p> <p>paa;</p> <p>purenta;</p> <p>hampaat;</p> <p>korvat;</p> <p>silmat;</p> <p>silmienvari;</p> <p>karvanvari;</p> <p>pigmentti;</p> <p>kaula;</p> <p>eturaajat;</p> <p>rintakeha;</p> <p>lantio;</p> <p>takaraajat;</p> <p>hanta;</p>		
------	--	---	--	--

		kapalat; luusto; karvapeite; selkalinja; vatsalinja; liikunta; kunto; lihaskunto; luonne; Tekstikenttä: kuvaus Sekä valinnat: aika paikka tarkastajat lopputulos <u>Aktiiviset nappulat:</u> Edellinen, Jatka, Peruuta		
--	--	---	--	--

4	Testaaja tarkastaa valintojen ja kenttien virheilmoitukset jättämällä lomakkeen valinnat ja kentät tyhjiksi ja klikkaa Jatka-painiketta	Sovelluksessa ei pääse eteenpäin, tarkistustulosten syöttölomake on sivulla auki ja kyseisten kenttien vieressä on virheilmoitukset		
5	Testaaja painaa Peruuta-painiketta	Avautuu Jalostustarkastuksen etusivu, jossa kolme linkkiä: <ul style="list-style-type: none"> - Tee tarkastus - Siirrä koirat hae koira		